UNITED STATES DISTRICT COURT
WESTERN DISTRICT OF WASHINGTON AT SEATTLE

THROUGHPUTER, INC., Plaintiff, v. MICROSOFT CORPORATION, Defendant.	COMPLAINT FOR PATENT INFRINGEMENT JURY DEMANDED

ThroughPuter, Inc. ("ThroughPuter") hereby alleges for its Complaint ("Complaint") against Microsoft Corp. ("Microsoft") as follows.

INTRODUCTION

1. A Field Programmable Gate Array (FPGA) is a specific type of microprocessor that can be reconfigured based on the tasks to be performed by it. In some situations, this reconfiguration takes place in between the performance of tasks by the FPGA. For example, the FPGAs can be configured and re-configured to provide hardware accelerators for data processing functions such as encoding/decoding, encryption/decryption or compression/decompression and then reconfigured to perform speech translation in addition to, *e.g.*, encryption and compression. Starting in 2015, Microsoft started putting FPGA processors in each server for Microsoft Azure

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

("Azure"), which is the world's largest cloud computing platform-as-a-service (PaaS). Using these FPGA processors for the underlying dynamic parallel execution environment or architecture, Microsoft claims a 150-200 fold improvement in data throughput and a 50 fold improvement in energy efficiency. Exhibit 15 at 91, Exhibit 29. Latency has also lowered by about a factor of 10. Exhibit 15 at 86, Exhibit 29. The end result is a power savings in Microsoft and increased processing speeds for Microsoft and the users of applications running on the world's largest cloud computing platform.

2. In 2013, Plaintiff ThroughPuter disclosed a reconfigurable and dynamic parallel execution architecture running on FPGA processors in writing to Microsoft. Two years later, Microsoft filed a patent application on the same hardware-based fabric (failing to disclose to the Patent Office any information about ThroughPuter's technology or earlier patent filings). Microsoft was ultimately awarded several patents on this subject matter. For example, in October 2020, Microsoft was awarded U.S. Patent No. 10,819,657, which claims that match almost exactly those of ThroughPuter's U.S. Patent No. 11,150,948, which is the basis for Count 1 of this complaint. In other words, Microsoft was awarded a patent on the same hardware-based fabric claimed in the ThroughPuter patents, wrongly suggesting that Microsoft had invented the architecture underlying Azure where ThroughPuter had already patented that architecture and disclosed it to Microsoft.

3. The figure below left is an excerpt from the materials disclosed by ThroughPuter to Microsoft in 2013, disclosing an application load and type adaptive "manycore fabric." The different colors show different reconfigurable cores (*e.g.,* logic blocks of FPGA processors) being assigned different tasks over time. The figure on the right is a colorized version of figures from

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

Microsoft's 2015 patent filing. The Microsoft figures show the same reconfiguration of cores to different tasks over time.

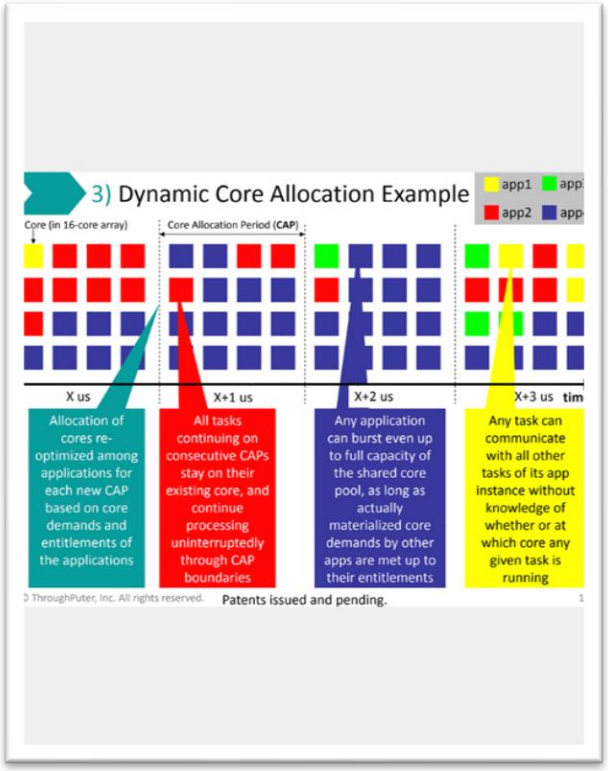
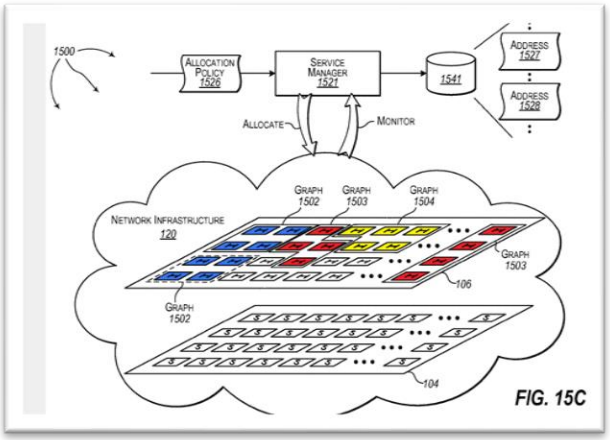


Exhibit 43 at 18.



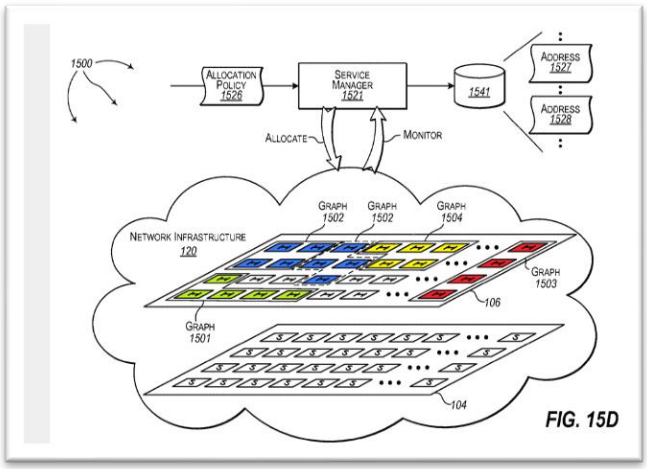


Exhibit 11, FIGs. 15C and 15D (annotated)

4. Microsoft's copying of ThroughPuter's technology was deliberate and wanton. While ThroughPuter has managed to survive as a small company developing and offering for sale other innovative solutions enabled by the throughput improvements which result from ThroughPuter's patented technology, Microsoft's copying and infringement of ThroughPuter's

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

intellectual property has irreparably damaged ThroughPuter's prospects of entering the PaaS market, which had grown from $3.8B in 2015 to $37.5B in 2019.[1]

5. On March 31, 2021, ThroughPuter filed suit against Microsoft in the United States District Court for the Eastern District of Virginia under Civil Action No. 3:21-cv-00216-MHL alleging infringement of related patents (the "Virginia Action"). ThroughPuter served the complaint in the Virginia Action on April 6, 2021. The Eastern District of Virginia transferred the Virginia Action to this Court by Order dated March 23, 2022. This Court assigned the transferred Virginia Action to Judge Rothstein as Civil Action No.2:22-cv-00355-BJR, which remains pending. This complaint asserts two patents that had not yet issued when ThroughPuter filed the Virginia Action.

6. Prior to filing the Virginia Action ThroughPuter repeatedly attempted to engage with Microsoft to resolve this matter outside of litigation. Microsoft rejected or ignored ThroughPuter's proposals and requests for business resolution of the matter.

7. Microsoft's decision to reject ThroughPuter's proposals and offers to collaborate created a highly inequitable situation by which Microsoft has used its tremendous market power to scale up the world's largest and most successful cloud computing platform. Microsoft did this based on the technology it copied from ThroughPuter: a small start-up that could have only competed based on its effort to protect and patent its innovations. Microsoft vaporized the competitive edge to which ThroughPuter was entitled by its decision to ignore ThroughPuter while

[1] https://www.statista.com/statistics/505248/worldwide-platform-as-a-service-revenue/

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

generating billions of dollars in revenue per quarter based on its unlawful exploitation of ThroughPuter's technology.

8. In a 2016 presentation entitled "Catapult at Ignite Innovation Keynote," Microsoft's Chief Executive Officer, Mr. Satya Nadella, boasted of the numerous benefits Microsoft had achieved by incorporating FPGAs into its cloud computing offerings: "We now have FPGA support across every compute node of Azure. That means we have the ability, ***through the magic of the fabric*** that we have built, to distribute your machine learning tasks, your deep neural nets, to all of the silicon that is available, so that you can get that performance, that scale." Exhibit 31 at 9 (emphasis added). The "magic of the fabric" did indeed allow Microsoft to scale up an FPGA-based cloud computing solution. But the "magic of the fabric" was not invented by Microsoft, it was copied from ThroughPuter.

9. According to Dr. Doug Burger (with whom ThroughPuter corresponded about its technology), the "magic of the fabric" referred to by Microsoft's CEO, use of FPGAs "allows us to do things on a scale that hasn't been done before." According to Dr. Burger: "It gives us the most powerful cloud, the most flexible cloud, and the most intelligent cloud." *Id.* at 30, 34.

10. The loss of ThroughPuter's technical competitive advantage through Microsoft's copying based infringement, which Microsoft admits gives it on the order of 100-fold performance and efficiency gain, has devastated ThroughPuter's business, including the ability to raise sufficient startup capital, gain collaborators, partners and initial customers, and to generally enter the market with a differentiated or more cost-efficient solution. ThroughPuter has survived notwithstanding Microsoft's anticompetitive tactics due to its ability to innovate in the fields of application of its core technology.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

11. ThroughPuter now brings this action based on Microsoft's willful infringement of the patents that form the causes of action herein.

NATURE OF THE ACTION

12. This is an action for patent infringement arising under the Patent Laws of the United States, 35 U.S.C. § 1 *et seq*., including 35 U.S.C. § 271.

13. ThroughPuter brings this action to halt Microsoft's infringement of its rights under the Patent Laws of the United States, 35 U.S.C. § 1 *et seq*., which arise under the following patents:

- U.S. Patent No. 11,50,948 (the " '948 patent), which is attached hereto as Exhibit 1 and
- U.S. Patent No. 11,036,556 (the " '556 patent), which is attached hereto as Exhibit 2[2].

THE PARTIES

14. Plaintiff ThroughPuter, Inc. is a Delaware corporation having a principal place of business at 249 Richmond Road, Williamsburg, VA 23185. Plaintiff owns over 50 issued domestic and foreign patents and pending applications protecting its products, services and technologies. ThroughPuter's President, Mark Sandstrom, is the named inventor on each of such patents and applications.

15. Plaintiff has developed and continues to develop various products and services, including i) Estimator™, a machine learning Application Specific Processor ("ASP")-as-a-service

[2] Exhibits 3-8 are intentionally left blank.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

offering of the ThroughPuter PaaS project, and ii) Grafword™, an artificial intelligence ("AI") powered, graphical authentication service that is a pilot application of the Estimator™ machine learning microservice.

16. Estimator™ provides a streaming machine learning ("ML") microservice, to support AI applications in unpredictably changing operating environments. Estimator™ allows its prediction models and logic parameters to be adjusted continuously while the microservice is in operation, such that its predictions will stay tuned-in to the prevailing reality of its operating environment, as that may evolve over time or even change abruptly. An International Search Report recently conducted by the International Search Authority under the Patent Cooperation Treaty concluded that this technology is patentable. A beta version of the Estimator™ application programming interface is currently commercially available for 3rd party developer subscription at www.estimatorlab.com.

17. Grafword™ provides graphic based high-security password generation and authentication, such that the level of authentication challenge is adjusted according to a level of deviation of a given user's online session attributes from what is expected for the given username. Grafword™ thus provides both high security as well as, for the authentic users, convenience in online authentication. An International Search Report recently conducted by the International Search Authority under the Patent Cooperation Treaty also concluded that this technology is patentable. A beta version of Grafword™ is used for Estimator™ account creation and login: https://estimatorlab.com/landing.

18. Both Estimator™ and Grafword™ have the potential to change the space in which they are offered due to the advantages provided by ThroughPuter's claimed inventions such as increased throughput and latency.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

19. Microsoft is a corporation organized and existing under the laws of the State of Washington with its principal place of business at One Microsoft Way, Redmond, WA 98052.

JURISDICTION AND VENUE

20. This is an action for patent infringement which arises under the Patent Laws of the United States, 35 U.S.C. § 1 et seq.

21. This Court has subject matter jurisdiction at least under 28 U.S.C. §§ 1331 and 1338.

22. Venue is proper in this District under 28 U.S.C. § 1400(b) because Microsoft has committed acts of infringement and has a regular and established place of business in this District.

23. This Court has personal jurisdiction over Defendant Microsoft pursuant to due process and/or Washington's Long Arm Statute because Microsoft has committed and continues to commit acts of patent infringement, including acts giving rise to this action, within the State of Washington and this District, and because Microsoft recruits Washington residents, directly or through an intermediary located in this state, for employment inside or outside this state.

24. The Court's exercise of jurisdiction over Microsoft would not offend traditional notions of fair play and substantial justice because Microsoft has established at least the required minimum contacts with the forum.

25. Microsoft maintains regular and established places of business throughout Washington and in this District.

26. Microsoft has substantial business contacts within this District and has purposefully availed itself of the privileges and benefits of the laws of the State of Washington.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

BACKGROUND

27. This case involves ThroughPuter's patented cloud computing, computing acceleration and related technologies, which were developed starting in 2010.

28. As of that time, advancements in computing technologies had generally fallen into two categories. First, in the field conventionally referred to as high performance computing, the main objective has been maximizing the processing speed of a given computationally intensive program running on dedicated hardware. In this field, speed was traditionally achieved by assigning a combination of separate parallel processors to all work on the same program simultaneously. Second, in the field conventionally referred to as utility or cloud computing, the main objective has been to most efficiently share a given pool of computing hardware resources among a large number of client application programs.

29. Thus, in effect, one branch of computing innovation has been seeking to effectively use a large number of parallel processors to accelerate execution of a single application program by parallelizing its processing across a maximum possible number of processors. At the same time, another branch of computing innovation has been seeking to share a single pool of computing capacity among a large number of application programs to optimize utilization of processing capacity. The former efforts pursue maximizing processing speed of a single program. The latter efforts pursue maximizing utilization of processing capacity.

30. As of the time of ThroughPuter's pioneering patent filings starting in 2011, there had not been major synergies between the effort to increase processing speed of a single program on the one hand, and maximizing processing capacity utilization on the other. Indeed, pursuing one of these traditional objectives often happened at the expense of the other, placing the two objectives in tension with each other.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

31. For instance, while dedicating an entire parallel processor based (super) computer to each individual application would increase processing speed of the individual programs, it would also cause severely sub-optimal computing resource utilization, as much of the capacity would be idle much of the time. On the other hand, while seeking to improve utilization of computing systems by sharing their processing capacity among a number of applications would lead to enhanced resource utilization, it also tended to slow down processing of individual programs. As such, the overall cost-efficiency of computing was not improving as much as improvements toward either of the two traditional objectives would imply: traditionally, increases in processing speed came at the expense of system utilization efficiency, while overall system utilization efficiency maximization came at the expense of individual application processing speed.

32. The foregoing tension was exacerbated by the fact that even mainstream application performance requirements were increasingly exceeding the processing throughput achievable from a single CPU core, *e.g.*, due to the practical limits being reached on the CPU clock rates. This created an emerging requirement for intra-application parallel processing (at ever finer grades) even for mainstream programs in order to pursue satisfactory processing speeds, while these programs were to be increasingly hosted on cloud platforms where the processing resources would be shared among programs of multiple clients.

33. These internally parallelized and/or pipelined (concurrent) enterprise and web applications would ultimately be largely deployed on dynamically shared cloud computing infrastructure by entities such as Microsoft using the technologies patented, pioneered and promoted by ThroughPuter.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

34. Given the foregoing, there existed a need as of 2011 for supporting a large number of concurrent applications on dynamically shared parallel processing resource pools. This then-existing need for a new parallel computing architecture could be met by a system that enabled increasing the speed of executing application programs (including through execution of a given application in parallel across multiple processor cores and/or using hardware accelerators) while at the same time improving the utilization of the available computing resources.

35. To address these problems, ThroughPuter developed hardware implemented dynamic resource management functionality including a scheduler, placer, inter-task communications and input/output system for use with multicore processor arrays dynamically shared among multiple concurrent applications, preferably to be deployed on FPGA processors. To that end, in this technology approach, the manycore processor array involves a fabric of reconfigurable cores that can be on-demand programmed to supply the needed mix or match of hardware accelerators. ThroughPuter's technology provided a cloud computing solution that enables accelerated processing speeds across multiple application programs while at the same time optimizing processing resource utilization.

36. An exemplary embodiment of ThroughPuter's Dynamic Parallel Execution (DPE) Environment™ (sometimes referred to as "DPEE™" or "DPEE™ technology") is depicted in, for example, ThroughPuter's U.S. Patent No. 9,424,090. Exhibit 9. Referring to the figures and tables of the representative '090 patent reproduced below, this preferred embodiment includes an **array of core slots 120**. The processors, or cores, are coordinated and coupled together with a **hardware-based manycore fabric managed by the controller**, which allows the client program tasks to be dynamically placed on processor cores of many-core arrays while communicating with each other directly and securely.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

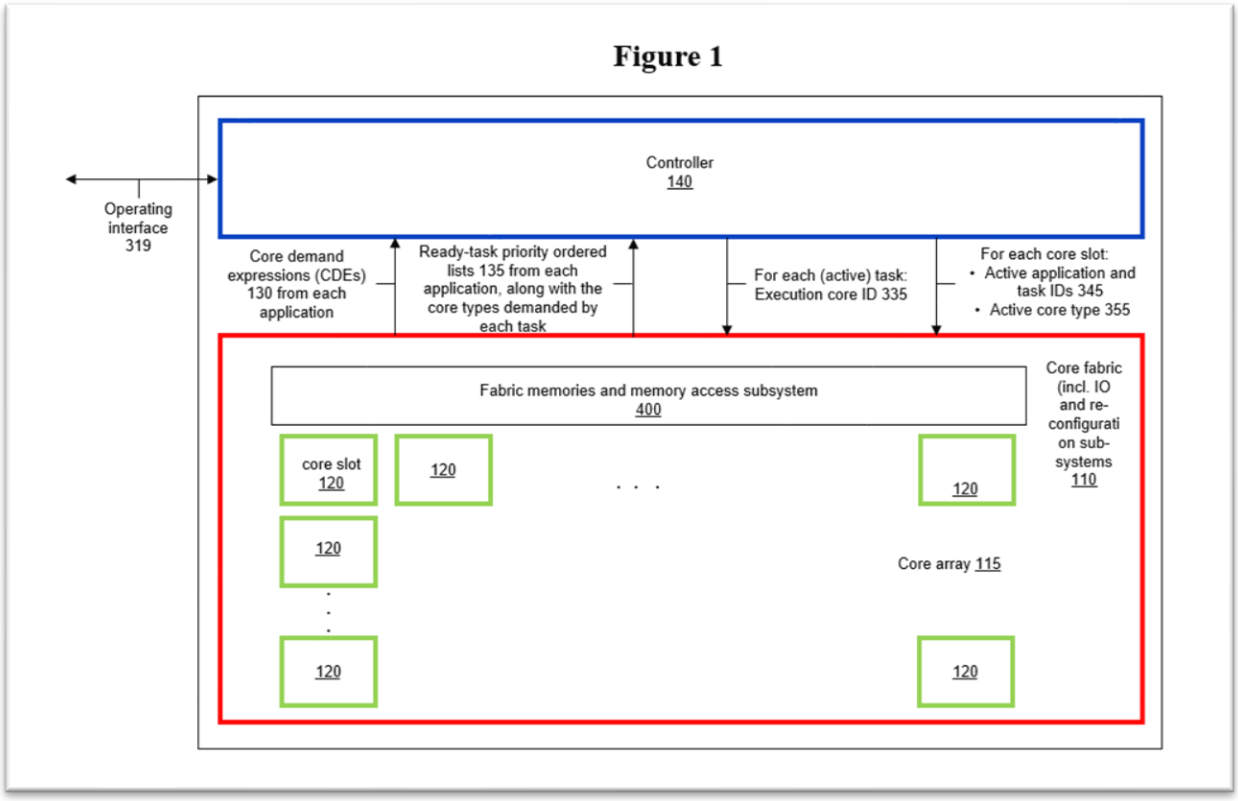


Exhibit 9, FIG. 1 (annotated)

37. ThroughPuter's hardware-based manycore fabric enables processing to be dynamically parallelized and hardware-accelerated, which results in optimized on-time processing throughput across the programs sharing an array of manycore processors. The effect for the client or end user of an accelerated service is increased processing speed and reduced cost base for delivering the application service, such that it becomes economically feasible for cloud service providers to support a range of performance intensive applications even without charge to end-users.

38. In a preferred embodiment, the **controller 140** monitors the processing load for each program sharing **the manycore array 110**, so that periodically re-optimized sets of tasks from the programs can be periodically assigned for execution on the **pool of processor cores 120**. Such periodic reassignment allows for optimal utilization of a pool of processing resources across

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

a number of potentially competing, concurrent applications. ThroughPuter's exemplary embodiment enables assigning time-variable sets of application tasks for execution on the fabric of reconfigurable cores.

39. A variety of **application tasks 240** are assigned to be run on the **core fabric**, which includes **an array of core slots** within the **core fabric**. Each **application task** may be assigned to run on a **single processor core** within the **core fabric**. The cross-connect, which connects the various components of **the hardware-based manycore fabric,** is used by the **controller** to repeatedly optimize assignment of **tasks** to the appropriate **processor core slots**.

Figure 2

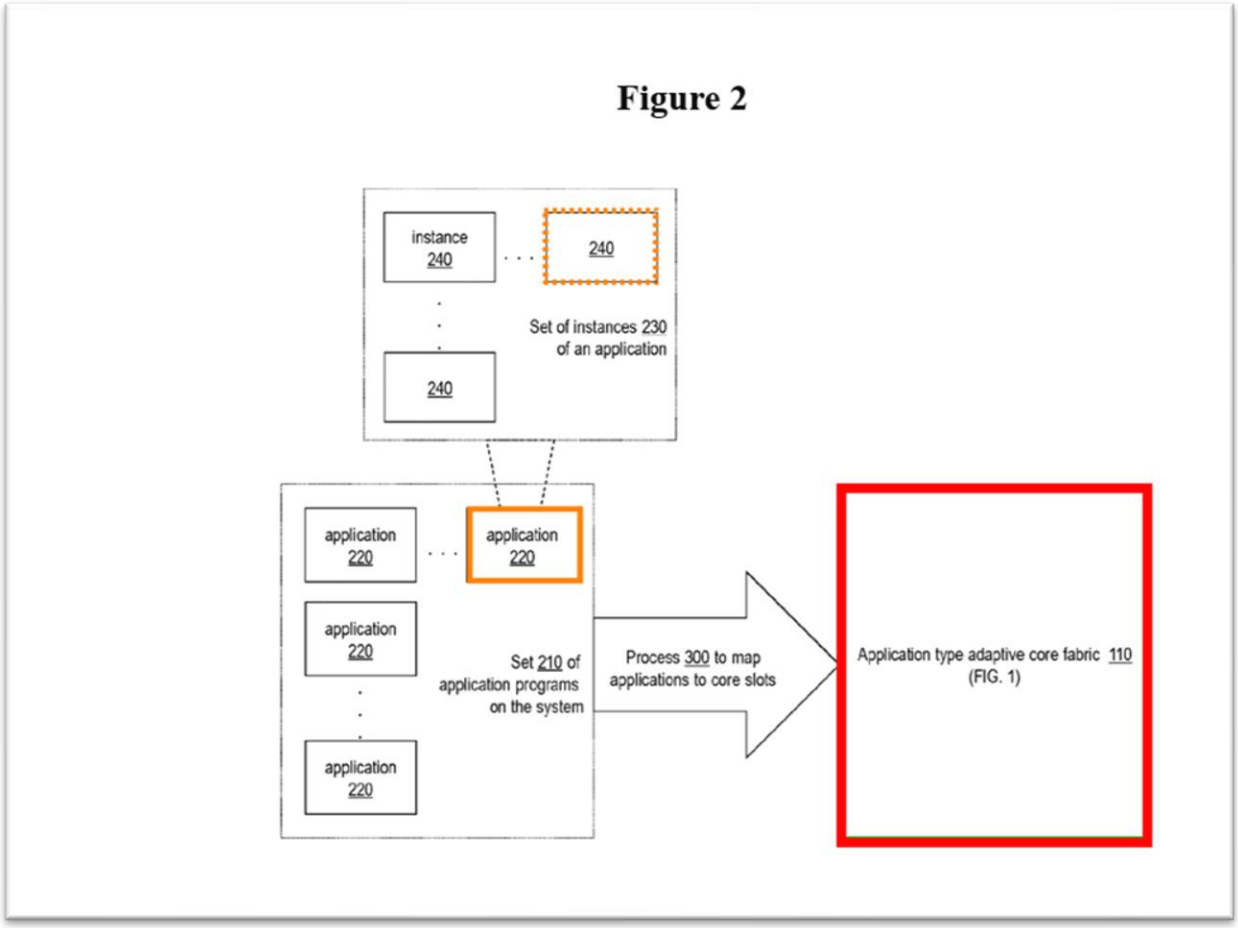


Exhibit 9, FIG. 2 (annotated)

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

40. The **controller** manages the **mapping** of **application tasks** to **processor cores** within **the core fabric**. The result is a system which dynamically and in real-time adapts to varying application processing loads to provide scalable, secure, high-performance and resource-efficient parallel cloud computing.

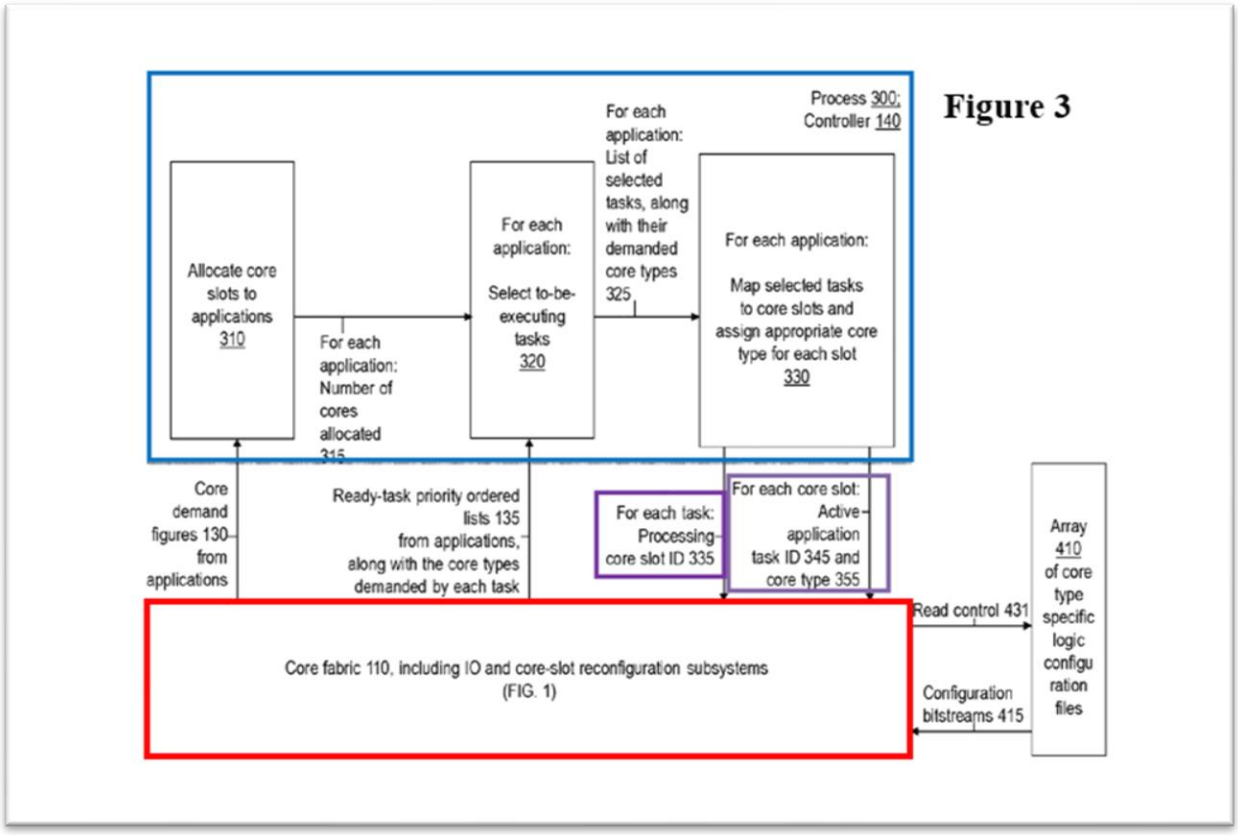


Exhibit 9, FIG. 3 (annotated)

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

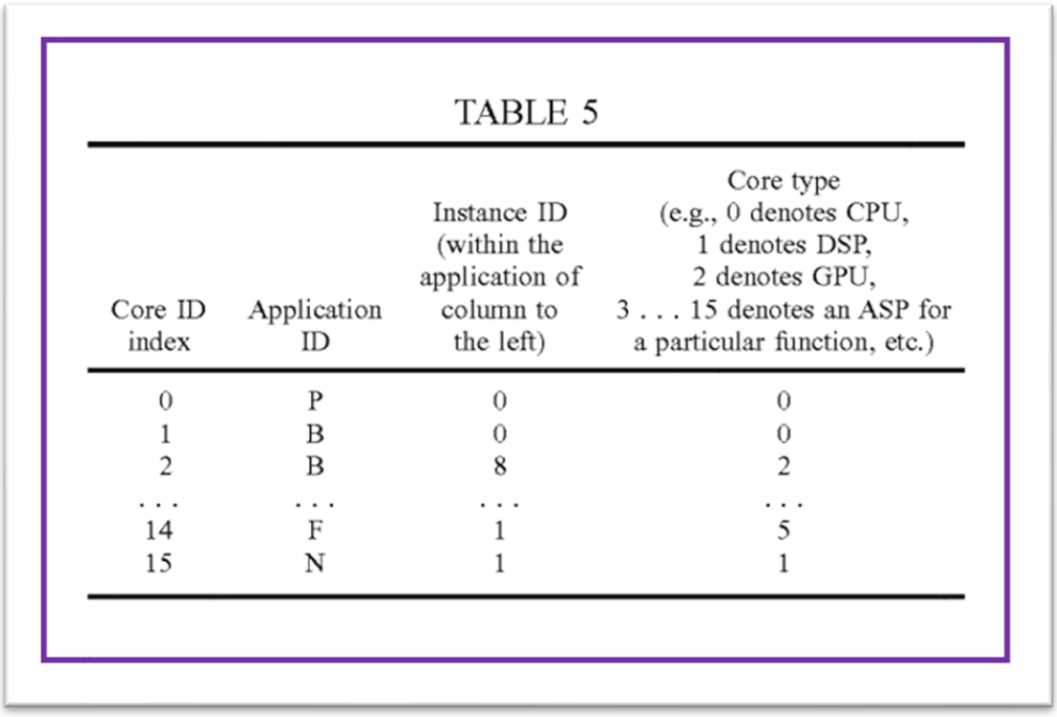


TABLE 5

Core ID index	Application ID	Instance ID (within the application of column to the left)	Core type (e.g., 0 denotes CPU, 1 denotes DSP, 2 denotes GPU, 3 . . . 15 denotes an ASP for a particular function, etc.)
0	P	0	0
1	B	0	0
2	B	8	2
. . .	. . .	. . .	. . .
14	F	1	5
15	N	1	1

Exhibit 9, Table 5 (annotated)

41. This novel application load and type adaptive manycore fabric permits tasks or applications to be managed at a high degree of granularity with minimized processing overhead while providing various advantages not previously attainable.

42. In recognition of ThroughPuter's innovative achievements, ThroughPuter's Mark Sandstrom was invited to speak at various high performance and cloud computing conferences starting in 2012. GigaOm selected ThroughPuter as one of eleven finalists to present at Launchpad 2012 in San Francisco, CA. That same year, ThroughPuter was invited to present its Dynamic Parallel Execution Environment (DPEE) based PaaS approach at the high performance computing start-up showcase at the Supercomputing 2012 conference ("SC12") in Provo, UT.

43. In February 2013, Mr. Sandstrom reached out to Dr. Doug Burger, Director of Client and Cloud Applications at Microsoft Corporation via email indicating that ThroughPuter

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

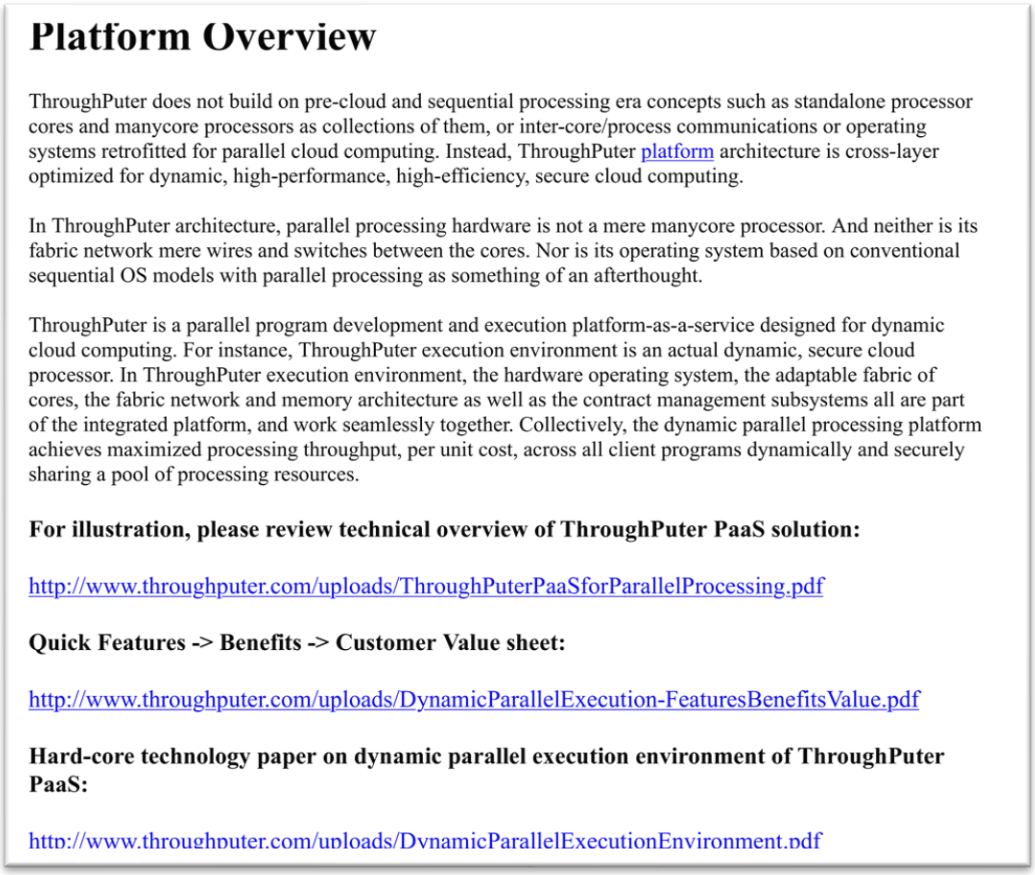
was looking for collaborators. The body of the email, which is attached as Exhibit 19, is reproduced below:

> Doug,
>
> I read at http://www.hpcwire.com/hpcwire/2013-02-01/kalray_produces_supercomputer-on-a-chip_for_industrial_applications.html your remarks at HiPEAC, concerning the need to modernize the computing architectures in view of the physical limits and future application requirements. In this context you may find ThroughPuter's cross-layer optimized platform architecture, based on dynamic parallel execution model, quite relevant -- please review http://www.throughputer.com/platform.html
>
> ThroughPuter is looking for collaborators for the effort to make the advanced dynamic parallel program execution capabilities available for users (application developers) via PaaS model; possibly Microsoft Azure business unit might be interested in exploring the collaboration opportunities.
>
> Feel welcome to share this call for collaboration among the appropriate parties at Microsoft, and naturally please get back to me for any questions etc. further discussions.

44. Dr. Burger responded "Thanks Mark ... I appreciate the note. I'll forward to the right people in Azure." *Id.*

45. In February 2013, the webpage at http://www.throughputer.com/platform.html had the following content at the time of the email exchange:

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301



Platform Overview

ThroughPuter does not build on pre-cloud and sequential processing era concepts such as standalone processor cores and manycore processors as collections of them, or inter-core/process communications or operating systems retrofitted for parallel cloud computing. Instead, ThroughPuter platform architecture is cross-layer optimized for dynamic, high-performance, high-efficiency, secure cloud computing.

In ThroughPuter architecture, parallel processing hardware is not a mere manycore processor. And neither is its fabric network mere wires and switches between the cores. Nor is its operating system based on conventional sequential OS models with parallel processing as something of an afterthought.

ThroughPuter is a parallel program development and execution platform-as-a-service designed for dynamic cloud computing. For instance, ThroughPuter execution environment is an actual dynamic, secure cloud processor. In ThroughPuter execution environment, the hardware operating system, the adaptable fabric of cores, the fabric network and memory architecture as well as the contract management subsystems all are part of the integrated platform, and work seamlessly together. Collectively, the dynamic parallel processing platform achieves maximized processing throughput, per unit cost, across all client programs dynamically and securely sharing a pool of processing resources.

For illustration, please review technical overview of ThroughPuter PaaS solution:

http://www.throughputer.com/uploads/ThroughPuterPaaSforParallelProcessing.pdf

Quick Features -> Benefits -> Customer Value sheet:

http://www.throughputer.com/uploads/DynamicParallelExecution-FeaturesBenefitsValue.pdf

Hard-core technology paper on dynamic parallel execution environment of ThroughPuter PaaS:

http://www.throughputer.com/uploads/DynamicParallelExecutionEnvironment.pdf

46. The third linked PDF entitled *Parallel Program Execution in a Dynamically Shared Adaptive Manycore Processor* is a document that closely follows the detailed description of ThroughPuter's '090 patent. Exhibit 20.

47. The second linked PDF entitled *Dynamic Parallel Execution via PaaS* is a document that provides an overview of the innovations described in ThroughPuter's portfolio of patents and applications including the application leading to the '090 patent and the substantial advantages they provide. Exhibit 21.

48. The first linked PDF entitled *Parallel Computing Development and Hosting Platform as a Service* is a presentation that provides an overview of ThroughPuter's DPEE. Exhibit 22. As shown in one of ThroughPuter's slides, reproduced below, the cores are dynamically re-assigned to instances or tasks of different applications, which are coded by color.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

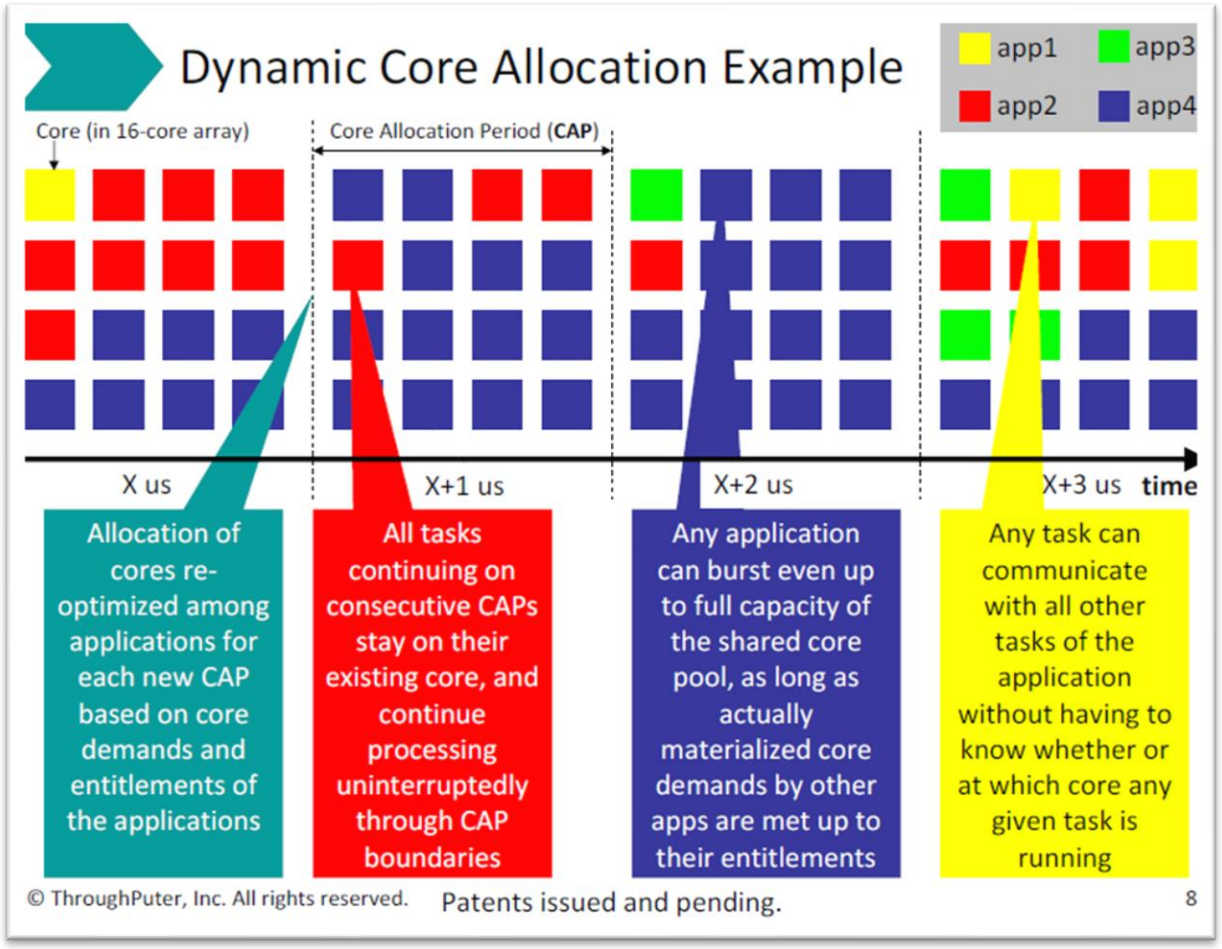


Exhibit 22 at 8

49. Repeatedly, the pool of (sixteen) processing cores are each allocated among the four different application programs. In an illustrative example, these may be application programs used for **speech translation (blue)**, **encryption (red)**, **search ranking (yellow)** and **data compression (green)**. In the first cycle or period, the search ranking **(yellow)** application program is allocated one core while the encryption **(red)** program application and speech translation **(blue)** application program are allocated eight and seven cores, respectively. For the next period, several of the cores are reassigned to the speech translation **(blue)** program application to accommodate an increase in processing demand made by that application. In the third period, a further core is allocated to the speech translation **(blue)** program application and the data compression **(green)**

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

program application is assigned one core, also in response to demand for those programs. In the fourth period, the cores are more evenly allocated across all four applications because the spike in demand for speech translation has abated, or because of demand spikes of the other applications. In this way, the processing throughput and latency across all the applications sharing the given manycore processor array are optimized. The end result for the application programs is faster processing compared to what would be achievable for equal cost base under non-adaptive capacity allocation or without on-demand acceleration.

50. ThroughPuter's novel manycore fabric led to industry recognition of ThroughPuter and its technology. For example, in January 2013, ThroughPuter was invited to publish an article in the Cloud Computing Journal, discussing the PaaS based on novel manycore fabric. Exhibit 12.

51. In addition, in September 2014, Mr. Sandstrom presented at the FPGAworld conference in Stockholm, Sweden, on the topic of *Hardware Implemented Scheduler, Placer, Inter-Task Communications and IO System Functions for Manycore Processors Dynamically Shared among Multiple Applications*. Exhibit 43.

52. By the time of the 2014 FPGAworld conference, ThroughPuter had already been granted at least a dozen U.S. and United Kingdom patents protecting techniques enabling the advantages of its Dynamic Parallel Execution Environment™ (DPEE).

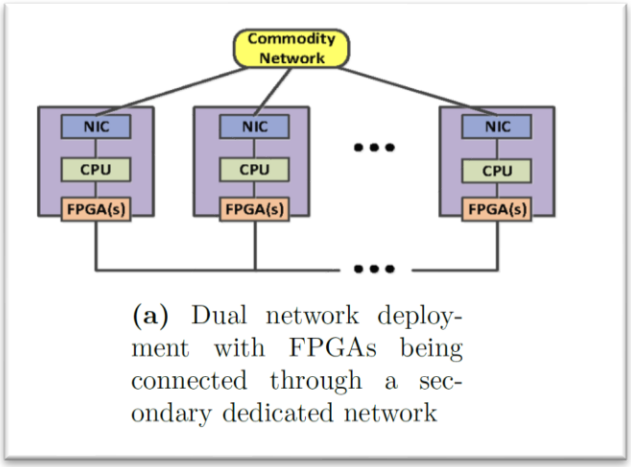
53. The following year, ThroughPuter was invited to present at the 2015 HPC Advisory Council Conference in Spain on the topic of executing multiple dynamically parallelized programs on dynamically shared cloud processors. A copy of the presentation is attached hereto as Exhibit 24.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

MICROSOFT'S INFRINGING CLOUD COMPUTING ARCHITECTURE

54. Microsoft's infringing cloud computing platform is known as Azure which is the technology service platform accused of infringement herein.

55. In April 2014, Microsoft introduced an early version of the Microsoft Azure reconfigurable hardware fabric in the whitepaper *A Reconfigurable Fabric for Accelerating Large-Scale Datacenter Services*, authored by Andrew Putnam, Adrian Caulfield, Eric Chung, Derek Chiou, Paolo Costa, Jeremy Fowers, and seventeen additional Microsoft engineers. Ex. 13. Microsoft initially called this reconfigurable fabric Catapult™ and still uses that term today. *Id.* at 1. This early version of Catapult, depicted in the figure at right, was deployed on a 1,632 server testbed. *Id.*, *see also* Ex. 40at 28-30.



(a) Dual network deployment with FPGAs being connected through a secondary dedicated network

56. The CPU and FPGA in each node communicate via PCIe. *Id.*, *see also* Ex. 35*passim*, Ex. 40at 28-30. *Id.* For external connectivity, dual network interfaces was used. *Id.* The first was Ethernet-based, which enables CPUs to communicate with each other over the commodity data center network. *Id.* The second was a specialized, low latency network implemented using 10Gb SAS cables which connects FPGAs to each other. *Id.* The topology for the second network was a 2D torus. *Id.*

57. The figure below provides details of components within the FPGA. *Id.* Here, custom load logic was implemented within an FPGA fabric allocation called "Role," while the Shell logic was composed of:

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

i) PCIe Core: This was used to implement an interface between the host CPU and the corresponding FPGA.

ii) SerialLite III (SLIII): This was a lightweight protocol used for inter-FPGA communication over the SAS links. Since each FPGA connects to four other FPGAs (2D Torus), there were four SLIII blocks in the Shell.

iii) DRAM Controllers: These were used to provide on-chip memory access to the Role or host (for DMA over PCIe).

iv) Inter-FPGA Router: A crossbar connected the four SLIII blocks, PCIe Core and Role.

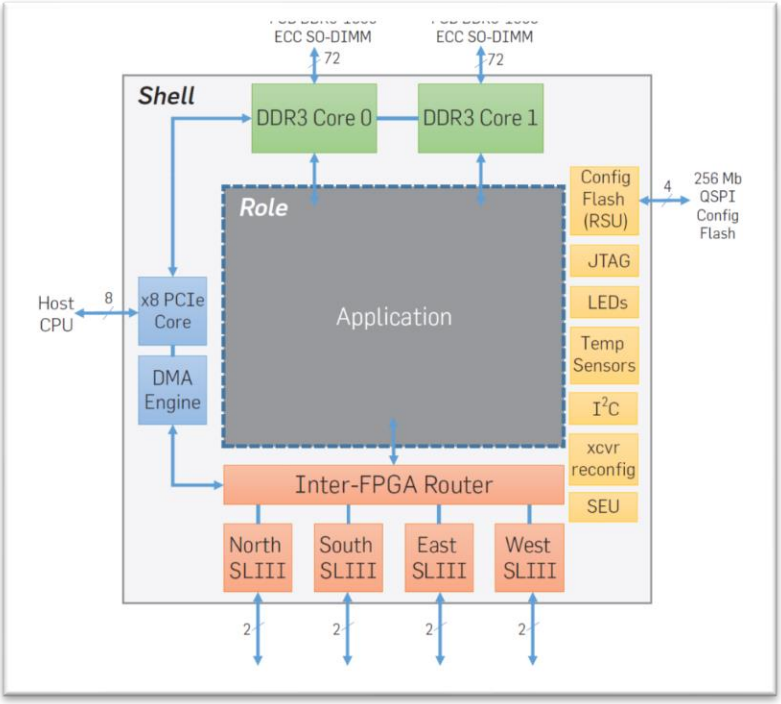


Exhibit 34 at 4; Ex 35 at 5

58. The foregoing version of Catapult will hereafter be referred to as Catapult I.

59. Circa 2015 Microsoft improved the Catapult I architecture to upgrade its initial scope of inter-FPGA connectivity, which permitted only communication between FPGAs in the same rack.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

60. Whereas Catapult I connected the FPGAs with "complex cabling" that "required awareness of the physical location of machines [i.e., each FPGA]" and provided only for inter-FPGA communication within a single rack (Ex. 14 at 1), the improved version includes a reconfigurable fabric enabling any FPGA in a datacenter to communicate with any other FPGA in the datacenter through super low latency multiplexer-switched connections. *See* Ex. 45(discussing the basics of how a multiplexer serves as a low latency, high speed switch), Ex. 29 at 148, 151.

61. This second version of Catapult, called Catapult II hereinafter, has been deployed on Azure servers since the launch of Catapult II circa 2015. Ex. 16 at 6 (indicating deployment in Azure "compute servers beginning in 2015."); Ex. 10 at 1 (indicating deployment at "hyperscale in Microsoft product data centers worldwide" by October 2016), Ex. 14 at 54.

62. At a high level, the improvement of Catapult II involved transforming the network from being a network of interconnected CPUs (some of which might have local, hardwired FGPA acceleration, sometimes called "bolt-on" acceleration) to a network of interconnected FPGAs (each of which can pass packets along to CPUs that are located "behind" the FPGAs). Stated differently, the network was transformed from being a network of CPUs (as was conventional) to being a network of FPGA processors/accelerators (as described in ThroughPuter's patent filings, which were disclosed to Microsoft years earlier).

63. In one of its whitepapers on the topic Microsoft explained that "[a]lthough this change in the network design might seem minor" it is a "major advance" that made a "profound" "impact [on] the types of workloads that can be accelerated and the scalability of the [system]." Ex. 14 at 54. Microsoft explained that this design improvement goes "far beyond just an improved network design" and "can be seen as a fundamental shift in the role of CPUs in the datacenter."

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

Id. at 58-59. Indeed, the network was changed from a *network of CPUs* with local FPGAs to a *network of FPGAs* with local CPUs. *Id. passim*.

64. In a 2016 presentation entitled "Catapult at Ignite Innovation Keynote," Microsoft's Chief Executive Officer, Mr. Satya Nadella, boasted of the numerous benefits Microsoft had achieved by incorporating FPGAs into its cloud computing offerings: "We now have FPGA support across every compute node of Azure. That means we have the ability, ***through the magic of the fabric*** that we have built, to distribute your machine learning tasks, your deep neural nets, to all of the silicon that is available, so that you can get that performance, that scale." Exhibit 31 at 9 (emphasis added). The "magic of the fabric" did indeed allow Microsoft to scale up an FPGA-based cloud computing solution. But the "magic of the fabric" was not invented by Microsoft, it was copied from ThroughPuter.

65. According to Dr. Doug Burger (with whom ThroughPuter corresponded about its technology), the "magic of the fabric" referred to by Microsoft's CEO, use of FPGAs "allows us to do things on a scale that hasn't been done before." According to Dr. Burger: "It gives us the most powerful cloud, the most flexible cloud, and the most intelligent cloud." *Id.* at 30, 34.

66. As will be described further below, since the launch of Catapult II the system has continued to evolve in ways that gradually incorporate more and more of the finer details, features and functionality described in ThroughPuter's patents. Many functions that were performed on the CPUs in the early days of the Catapult II fabric have been gradually shifted to the FPGAs. The Azure fabric (sometimes referred to hereinafter as the "Catapult II network," the "current Catapult II network" or simply "Catapult II") strongly resembles or is nearly identical in relevant respects to the hardware-logic (e.g. FPGA) fabric disclosed in ThroughPuter's patents.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

67. Returning to the "major advance" that is embodied in the Catapult II reconfigurable fabric, central to that advance was the decision to permit every FPGA in a datacenter communicate with any other through a high-speed, low latency, multiplexer switched fabric as described in the ThroughPuter patents. Exs. 10, 15, 36 *passim*. The figure below shows the "shell" of the Catapult II FPGA. Exs. 10, 14. All of the depicted functionality is performed on the FPGA itself, providing high-speed, low latency interconnectivity. *Id.*, *see also* Ex. 46 *passim*, Ex. 16 at 5.

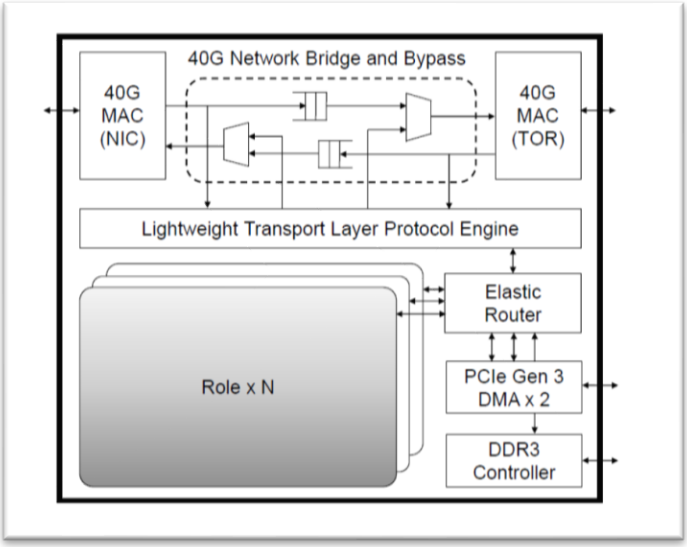


Exhibit 10 at 4

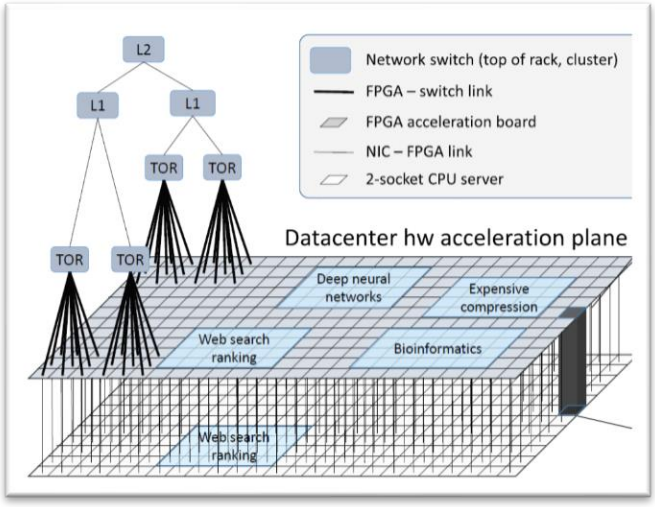
68. Via this shell, the Catapult II inter-FPGA communications and memory accesses occur through the high-speed, low latency multiplexer network inside the FPGA. *Id.* For example, the DDR3 controller comprises DMA multiplexers and demultiplexers. *Id.*, *see also* Ex. 36 at 3. As another example, the bridge between the Lightweight Transport Layer Protocol Engine and the ports for the NIC and TOR includes several functional multiplexer units.

69. In Catapult II, about 23% of each FPGA's logical circuitry (again, comprising logic units interconnected by high-speed, low latency multiplexer networks) is used to provide the new functionality, including the elastic router, LTL engine, and bridge and the other adapters and

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

controllers of Catapult II. *Compare* Ex. 14 at 4 to Ex. 35 at 5. Collectively, these components (hereinafter "Catapult II Shell Components") provide and facilitate FPGA fabric interconnectivity and memory access. Exs. 10, 14, 36 *passim*.

70. Each Azure rack in a datacenter is connected by a two-tier switching network L1/L2 as shown below. Exs. 10, 14, *passim*. The packet communications within each rack and to the top-of-rack switch (TOR) are facilitated by the FPGA via the high-speed, low-latency multiplexer controlled connections that serve as the Catapult II Shell Components. *Id.*



71. Upon information and belief, since at least 2017, the TOR switch and the L1/L2 switches have included multiplexer-switched network connections to reduce latency. Ex. 38 at §4.1 ("4.1 Switch design . . . Our circuit switch operates at layer 1, i.e., data traversing the switch is routed through the PHY block at the ingress and egress ports (Fig. 6). ... The control signals to these multiplexers are driven by p registers, one per multiplexer. . . Hence, the switch reconfiguration delay is simply the time it takes to update the registers, which can be done in one clock cycle."), Ex. 39 at §4.1 (4.1 Switch Design . . . .This is implemented using multiplexers whose control signal is driven by registers storing the clock counter and timeslot. Therefore, the switch reconfiguration latency corresponds to the time required to reconfigure the multiplexers,

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

which is just a few logic gate delays and well below one FPGA clock cycle.")In 2016, Microsoft published a paper entitled *A Cloud-Scale Acceleration Architecture* authored by, among others, Adrian Caulfield (hereafter "Cloud-Scale Acceleration Architecture"). Exhibit 10.

72. Microsoft Azure includes or has included the functionality as described in Cloud-Scale Acceleration Architecture.

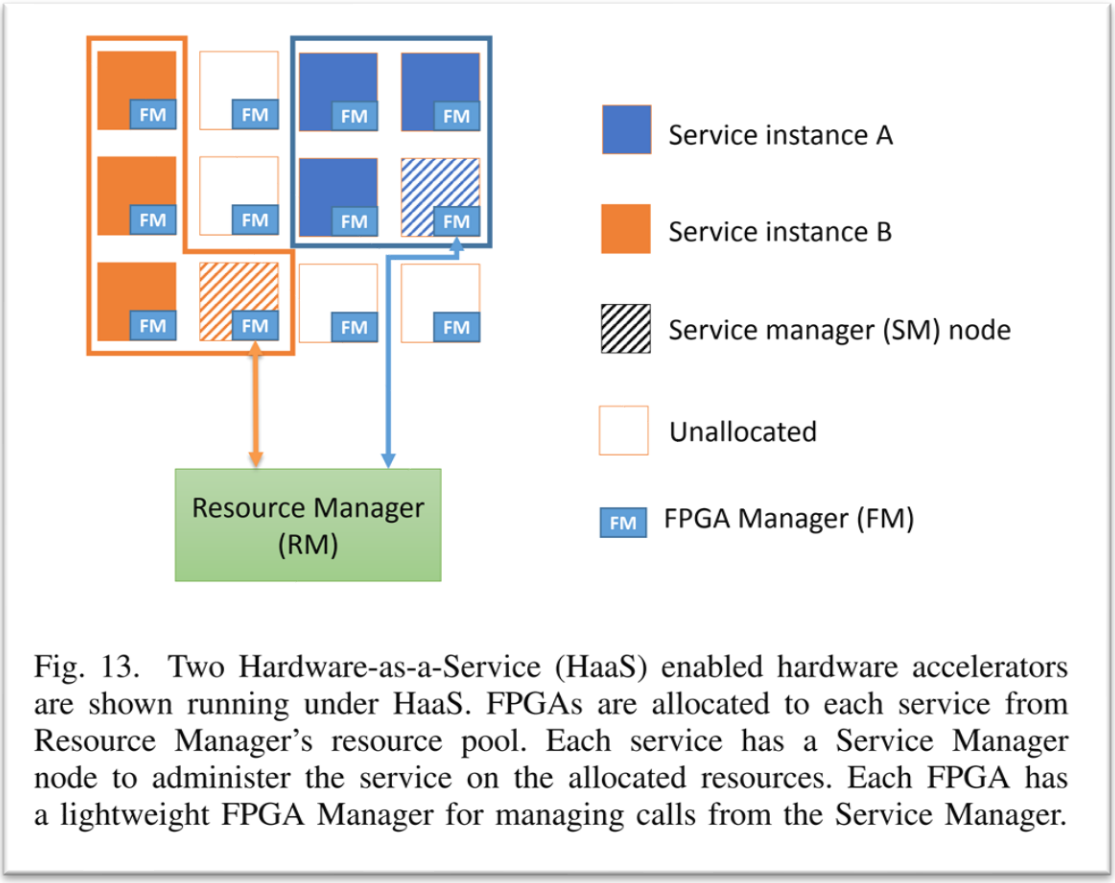


Fig. 13. Two Hardware-as-a-Service (HaaS) enabled hardware accelerators are shown running under HaaS. FPGAs are allocated to each service from Resource Manager's resource pool. Each service has a Service Manager node to administer the service on the allocated resources. Each FPGA has a lightweight FPGA Manager for managing calls from the Service Manager.

Exhibit 10, Fig. 13

73. Each Service Manager (SM) described in Cloud-Scale Acceleration Architecture is associated with a given, dynamically sized and placed, FPGA processor group, each of which is referred to as a "service instance" in Cloud-Scale Acceleration Architecture.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

74. Cloud-Scale Acceleration Architecture states that the functionality described therein is used to "accelerate . . . Azure infrastructure workloads," among other use cases. Exhibit 10 at 13.

75. Microsoft's presentation entitled *Inside Microsoft's FPGA-Based Configurable Cloud* (Exhibit 15) demonstrates that Azure embodies or has embodied the functionality described in Cloud-Scale Acceleration Architecture (Exhibit 10) and Microsoft's U.S. Patent No. 10,270,709 (the " '709 patent) (Exhibit 11).

76. As explained in the presentation, Microsoft Azure is a system comprised of a number of FPGA-accelerated servers, which process tasks from multiple application programs involving a controller that allocates FPGA processors from among a number of FPGA processors and then assigns application tasks to FPGA processors in order to pursue accelerated application processing while seeking efficient processing capacity usage.

77. A transcript and corresponding screenshots of Microsoft's presentation entitled *Inside Microsoft's FPGA-Based Configurable Cloud* are attached as Exhibit 15. The video is available at https://www.youtube.com/watch?v=v_4Ap1bjwgs.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

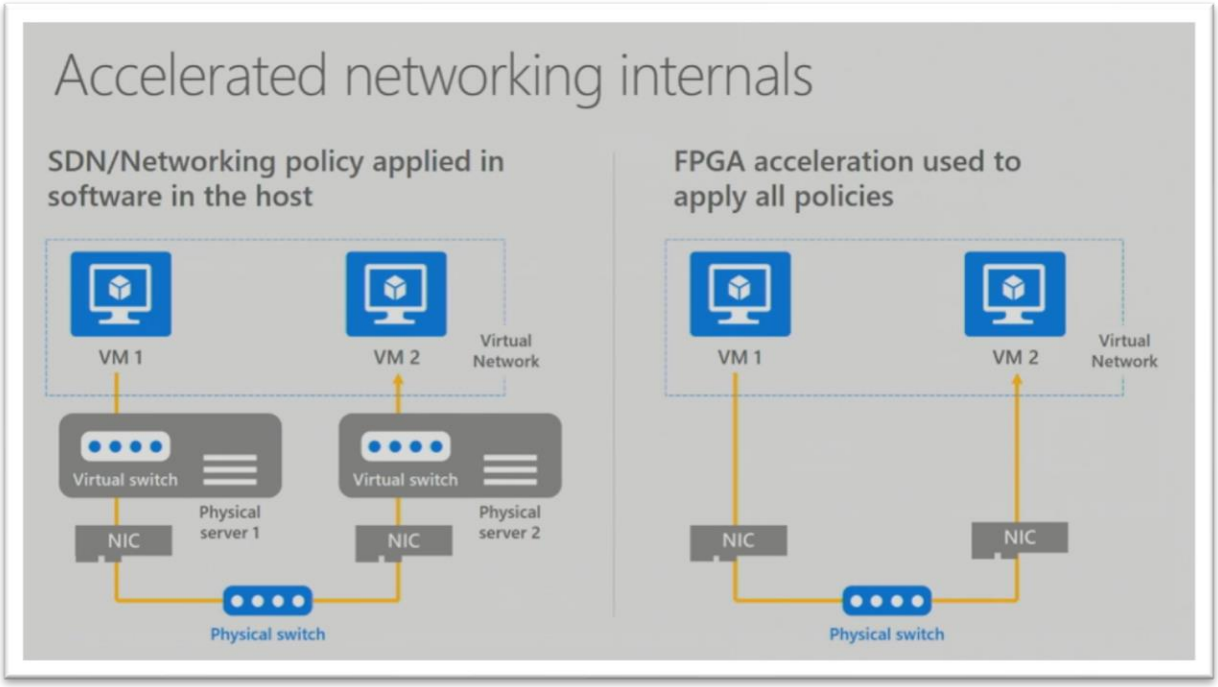
78. Microsoft Azure's RM interfaces with the Azure controller (AC) that applies policies to the data packets transmitted to and from the processors within Azure. Exhibit 15 at 39-42. The RM configures a hardware switch to execute the policies set forth in the hardware access control lists (ACLs), which list grant access rights to a given application program. *Id.*

79. Under the direction of the AC, load balancing rules are practiced to pursue optimization of processing capacity by defining how tasks are distributed to FPGA Virtual Machines (*e.g.*, FPGA groupings or graphs in the parlance of Microsoft's '709 patent) (hereafter "FPGA Groupings"). Exhibit 15 at 43-50.

80. The AC instructs the fabric how to route each data packet associated with an application program or service instance task to a given processor based on the type of task to be performed and the available processing capacity on the Azure resource pool. The AC sets up load balancing rules in the load balancing control table (LB NAT) and pushes them out to all servers hosting FPGA Groupings.

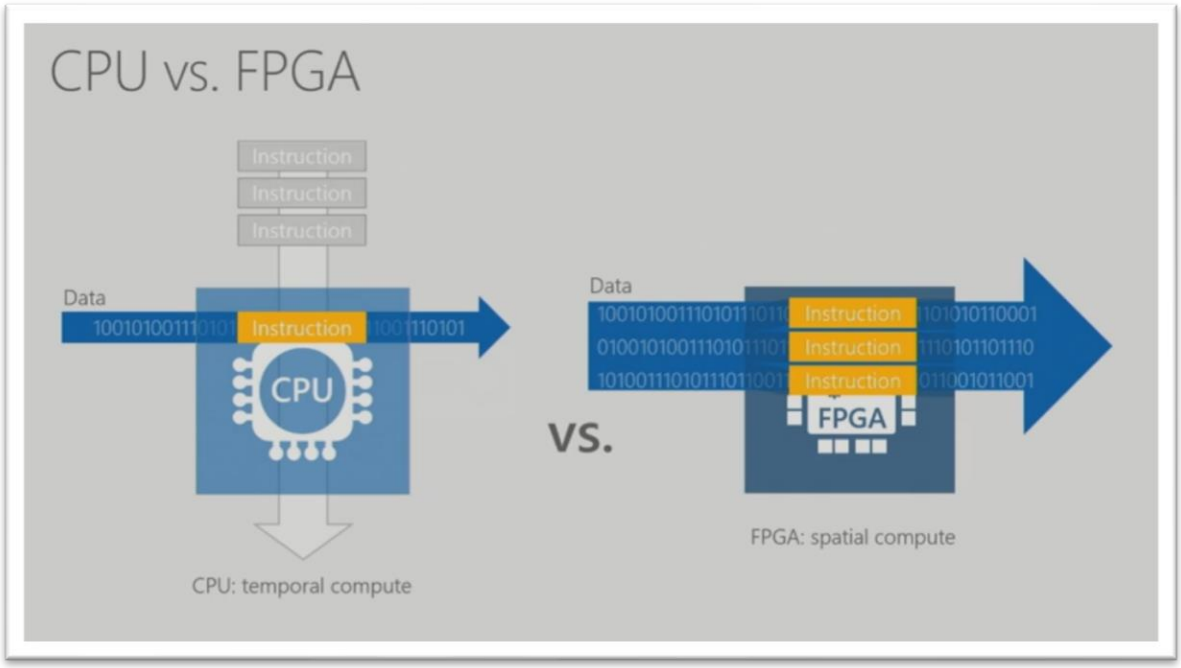
LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

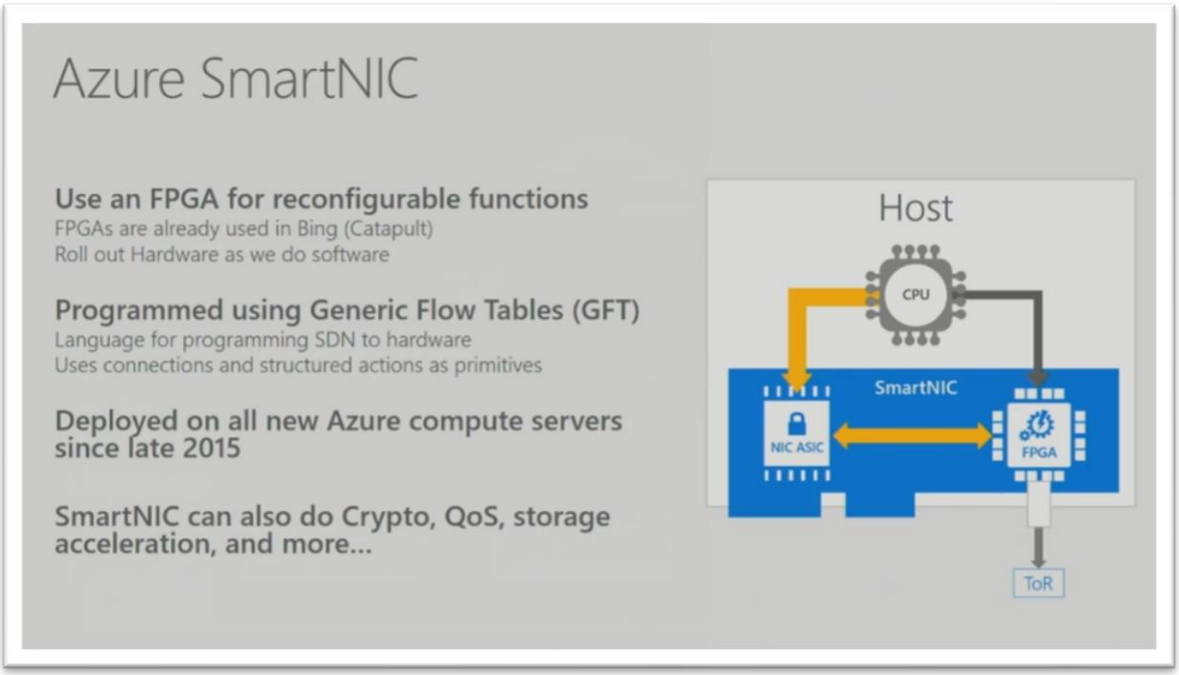
81. Under the direction of the AC, Microsoft Azure's physical switch connects directly to the network interface cards (NICs) to route packets (*e.g.*, messages) to application or service instance tasks running on the FPGA Grouping. Exhibit 15 at 54-57. Each FPGA Grouping consists of several FPGA processors operating in a coordinated fashion under the control of the RM in conjunction with an SM, and a FPGA Manager or Node Manager (FM), as explained further below.



82. The FPGA processors are reconfigured to perform various functions for different application programs. *See, e.g.*, Exhibit 15 at 19-24, 54. As explained further below, the FPGA processors are dynamically reconfigured to be suited for a given task (*e.g.*, deep neural network (DNN) application tasks, SQL database tasks, and so on).

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301



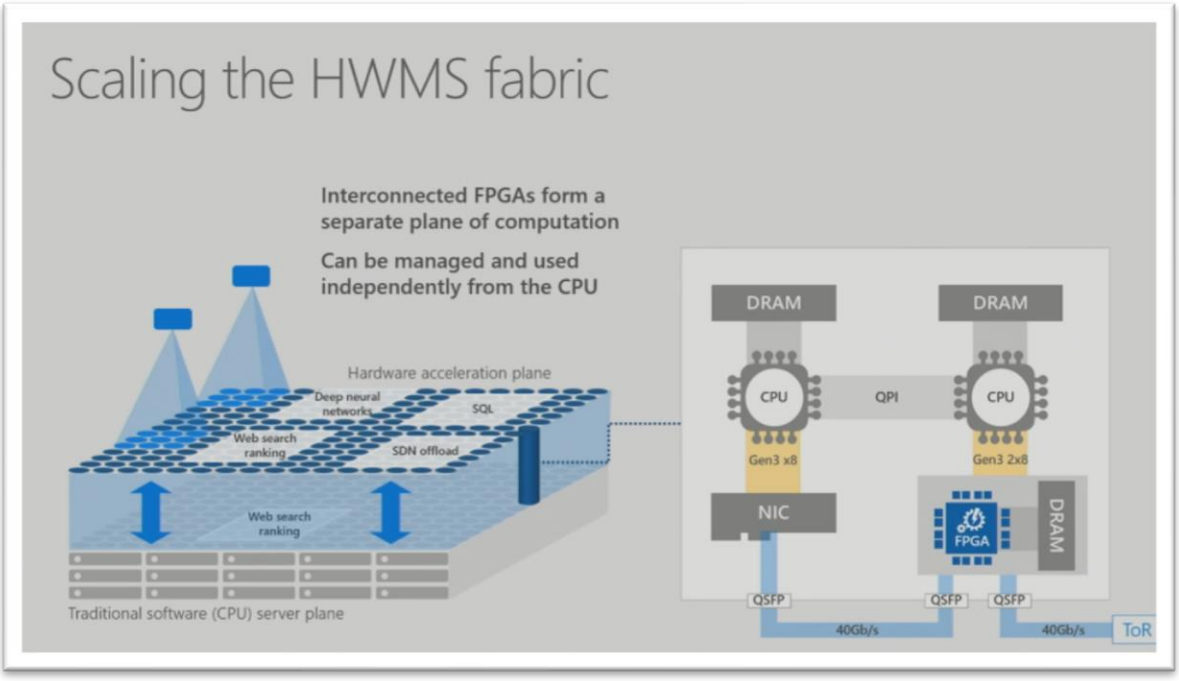


83. As shown in the above figure, an Azure Server includes an FPGA, a NIC, and a CPU. The FPGA is connected with the NIC and CPU, and the Azure Server is connected to other Azure Servers through a network, constituting a fabric of pooled FPGA accelerators. In an

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

implementation disclosed in Fig. 8 in the '709 patent, an Azure Server includes an FPGA subdivided into separate configurable domains. In this implementation, the separate configurable domains are locally connected to each other through on-chip/on-board hardware. This forms a fabric of accelerator units that may be, at least in part, internal to a subdivided FPGA (hereafter the fabrics referred to in this paragraph are referred to as "Azure Fabric").

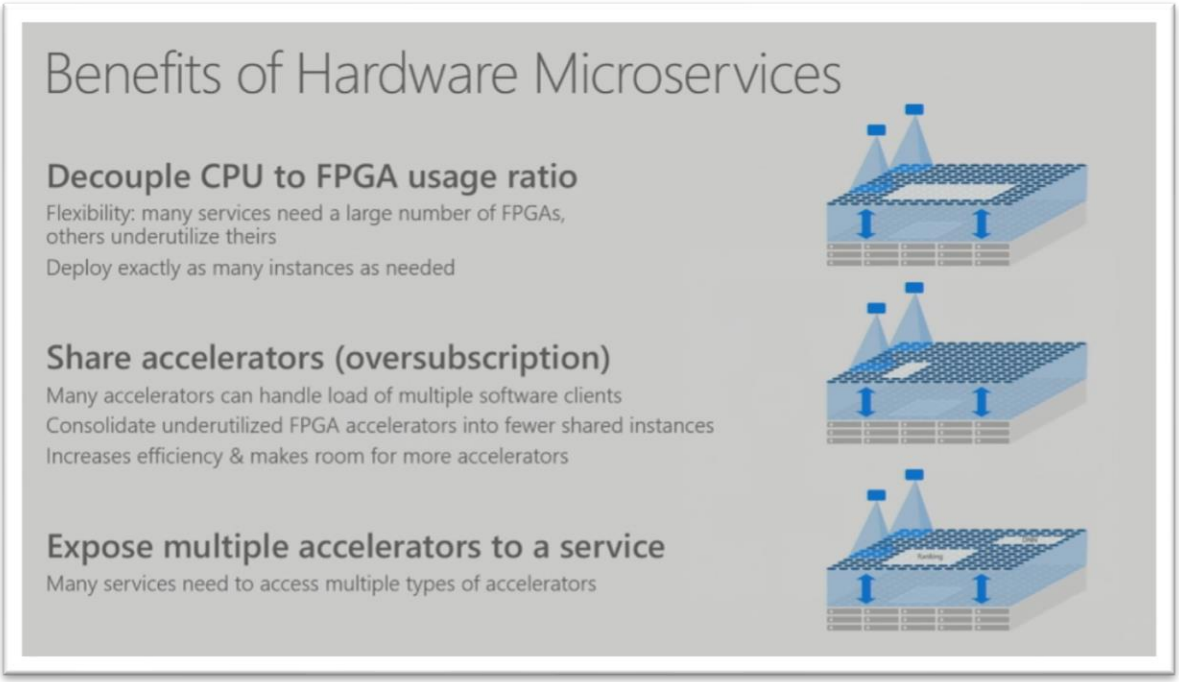
84. The FPGA processors are arranged in groups to execute tasks associated with various application programs. Exhibit 15 at 93-98. For example, one FPGA service instance or graph, *e.g.*, an FPGA Grouping, (the blue rectangle with downwardly projecting triangle) executes deep neural network (DNN) application tasks. *Id*. Another executes SQL database tasks, yet another executes network protocol offload tasks, and the last executes web search ranking tasks. Each FPGA Grouping includes several individual FPGA processors (illustrated as blue ovals in the "hardware acceleration plane").



85. In this manner, the AC repeatedly and dynamically rearranges task assignment to

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

the array of processing units (*e.g.*, FPGA processors) while rearranging communication path connectivity for the array of processing units to optimize the application processing performance as well as the usage of processing capacity on the Azure system. Exhibit 15 at 95-99. The FPGA Groupings (and thus the associated connectivity) changes over time in reaction to the processing demand of the various application programs.



86. Microsoft is now using this FPGA fabric architecture in every new server it deploys in its data centers. Exhibit 28 at 8. Using this architecture, Microsoft obtained 40-100 fold performance improvements in Microsoft Bing's machine learning algorithms. *Id.* Microsoft has also announced the availability of Brainwave, an FPGA-based system for ultra-low latency deep learning for Azure. *Id.*

87. Using this architecture, Microsoft claims up to a 150-200 fold improvement in data processing throughput and up to a 50 fold improvement in energy efficiency. Exhibit 15 at 91, Exhibit 29. Latency has also lowered by about a factor of 10. Exhibit 15 at 86, Exhibit 29.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

88. In Azure, the RM works in conjunction with both (a) the FM on each host processing unit, which keeps track of resource allocations on the FPGA processors, and (b) the SM, which assigns computing tasks to the FPGA processors. Exhibit 15 at 103-06.

89. In the implementation illustrated below, an SM in conjunction with the RM assigns one grouping of computing tasks of a ranking service application program (*e.g.*, Bing) to the FPGA processors that have been allocated. Another grouping of FPGA processors is performing tasks for the Azure Data Link Analytics (ADLA) application, with those tasks having been assigned by another SM in conjunction with the RM, and so forth. In this way, the RM, SM and FMs ensure that each FPGA processor is receiving data from the appropriate queue and executing a task corresponding to the proper application. The RM, SM and FMs also ensure that the output of each FPGA accelerator is communicated to the proper output queue. The load balancing table is utilized to allocate application loads among FPGA processors and FPGAs among loads in order to pursue optimal usage of available processing capacity. In the illustrated example, the Ranking Service is expressing higher processing demand and/or has higher prioritization resulting in it getting allocated four FPGA cores and/or FPGA Groupings, compared with *e.g.*, ADLA that is assigned two FPGA accelerators.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

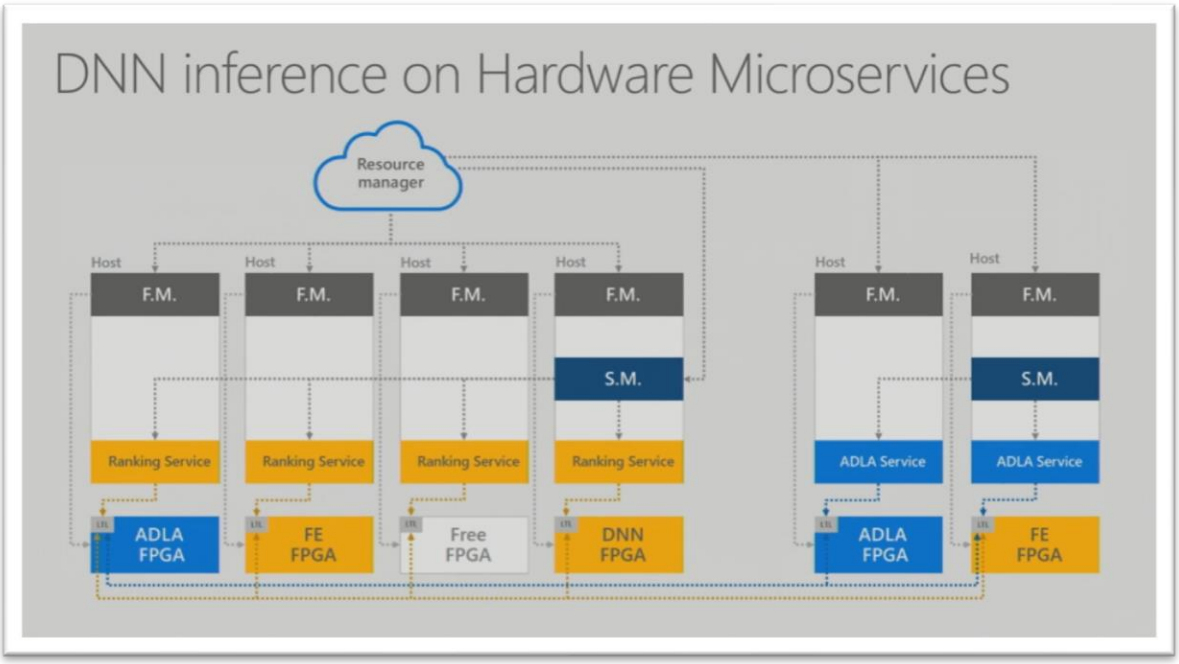


Exhibit 15 at 104

90. Azure's queuing of data as it moves between memories and the FPGA processors is or has been accordant to descriptions in Microsoft's United States Patent No. 10,296,392 ("the '392 patent"). Exhibit 47.

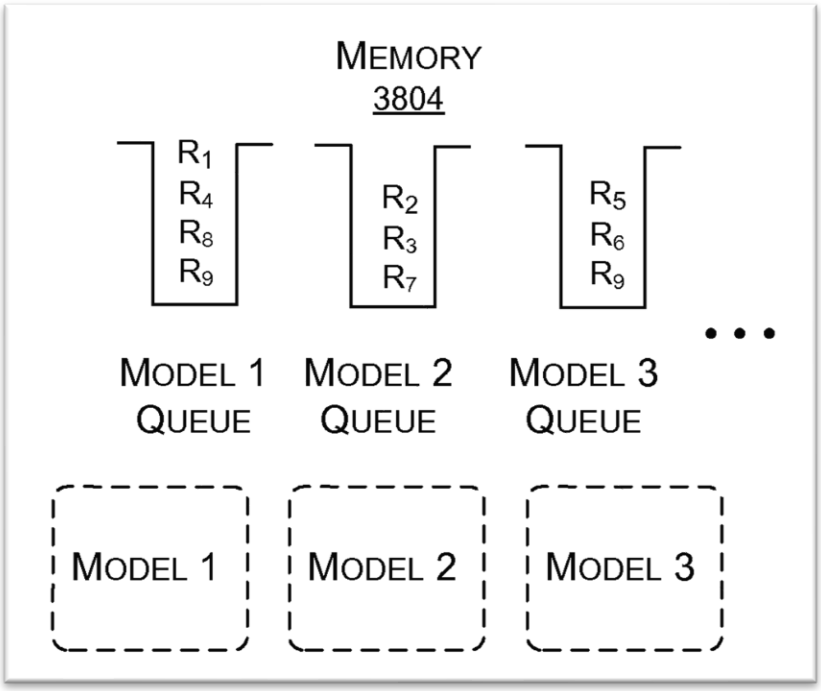
91. On information and belief, Microsoft Azure embodies or has embodied the functionality substantially as described in the '392 patent.

92. In connection with Fig. 38 (excerpted below), the '392 patent explains that different application programs such as a French language search engine, an English language search engine, and a German language search engine may have dedicated input buffers Model Queue 1, Model Queue 2 and Model Queue 3, respectively. Exhibit 47 at 34:62-35:34; see also Fig. 38. Data is fed from the queues to the FPGA processors based on policies such as queue fullness. *Id.* Generally, the queue manager (in coordination with at least the SM) will prioritize the processing of queries in the queue with the most queries. As an example, shown below, Model 1 Queue has

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

queued four queries (R_1, R_4, R_8 and R_9) where Model 2 and Model 3 Queues have queued three queries. As such, Model 1 Queue would be prioritized above the other two.

93. On information and belief, Azure uses such queues to manage the transmission of data to, from and/or within the FPGA processors.



94. The distribution of tasks among the FPGA processors in Azure by the SM, RM, and FMs is also substantially described in the '392 patent.

95. In operations 1-6 illustrated in Fig. 2 of the '392 patent, each hardware acceleration component 208/210 (which may be an FPGA or a subdivision thereof) performs a task on the data and forwards the result to the next destination under the management of the head component of the FPGA Grouping (also referred to as the SM). The '392 patent further explains in connection with Fig. 36 that each acceleration component performs its respective function, such as mathematically combining feature values or compressing the data reflecting the feature values computed thus far.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

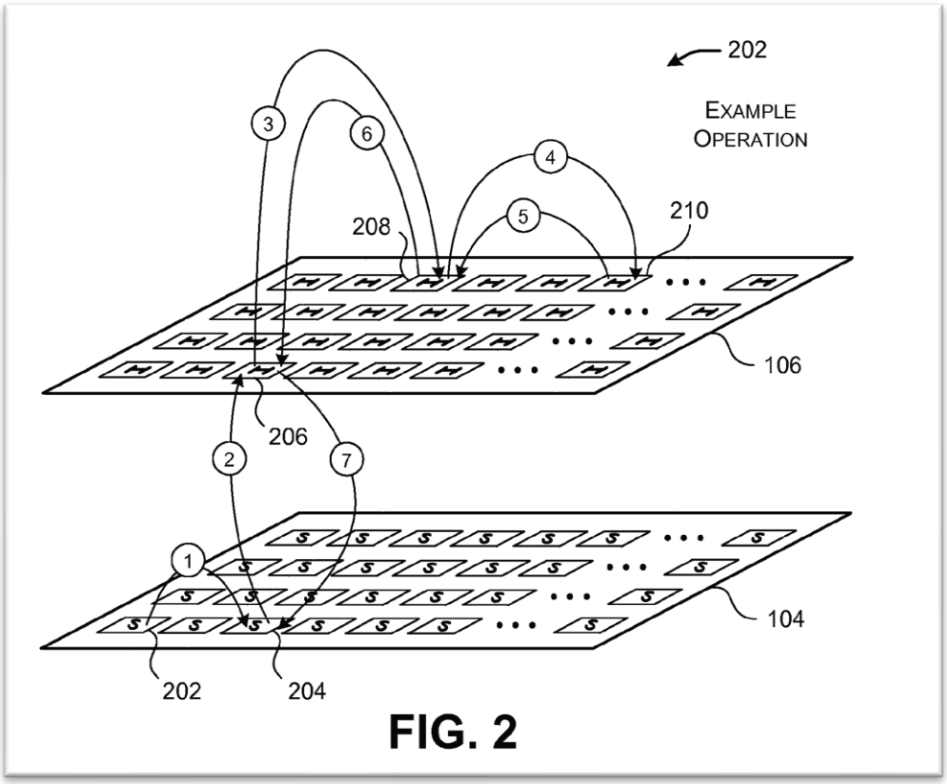


FIG. 2

96. The scheduling and placement of tasks by Azure through the RM, SMs, and FMs is done taking into consideration a given task's readiness for execution. Exhibit 47 at 35:20-25; Exhibit 17 at pp. 287-88.

97. On information and belief, Azure embodies or has embodied the technology described in Exhibit 17.

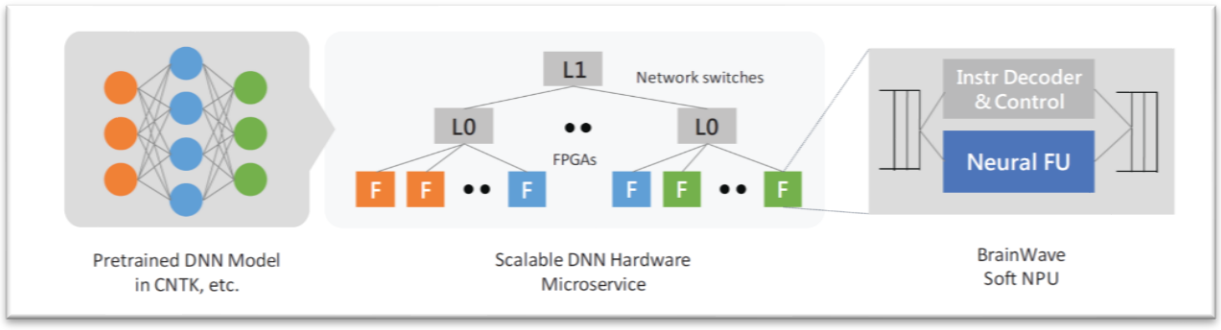
98. Microsoft Azure also supports assured resource allocations for a client's FPGA Grouping for an additional fee. *See* Exhibit 48 at p. 3. Azure allocates FPGA processors based in part on the number of FPGA Groupings subscribed to by the client for its applications.

99. In late 2017, Microsoft began moving its deep neural networks (DNNs), which provide the backbone for their real time artificial intelligence (AI) applications, from CPU server networks to Catapult II FPGA server networks. Exs. 18, 37*passim*, see also https://www.microsoft.com/en-us/research/ blog/microsoft-unveils-project-brainwave/. These

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

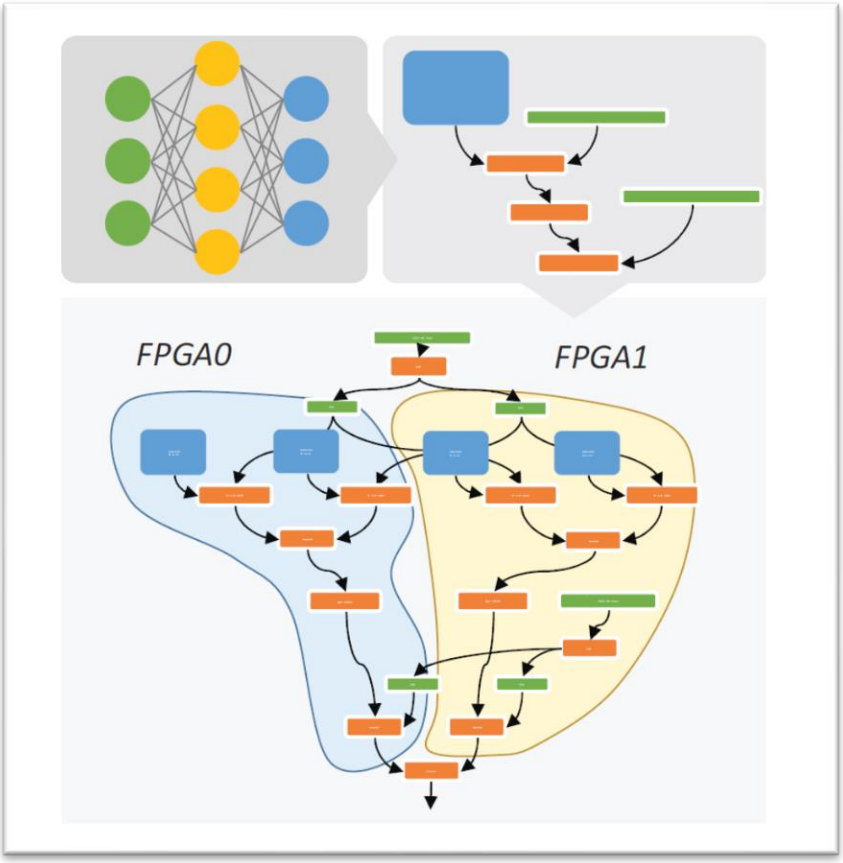
FPGA-enabled DNNs were used internally by Microsoft to support Bing and Azure in 2017 and became available to customers in 2018. Ex. 18 at 17. Microsoft uses the code-name Project Brainwave to refer to its platform that accelerates DNN inferencing, which runs onto of the Catapult servers in Azure.

100. Microsoft's DNN models deployed on FPGA servers (i.e., Brainwave deployed on the Catapult II FPGA network utilize the multi-stage architecture described in various ThroughPuter patent applications, which were published and/or issued as patents more than three years earlier. See, e.g., Ex. 8. As illustrated and explained in one of Microsoft's whitepapers on the topic (Ex. 18), the DNNs are implemented as a hardware microservice having multiple stages. The first stage (illustrated in orange) is implemented on a flexible number of FPGA processing cores and passes its intermediate processing result to the second stage (illustrated in blue), which in turn passes its result on to the last stage (illustrated in green), which outputs the DNN hardware microservice result to the calling application.



LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

101. The FPGA cores of the different FPGAs that are performing the same task constitute a stage. *Id.* In the illustration below (an excerpt of Ex. 18 Fig. 3), the tasks by the DNN model are separated into three types of tasks, illustrated in blue, orange and green. *Id.* Each of tasks is deployed on a FPGA processing core labelled FPGA0 and FPGA1. *Id.* The blue cores operate collectively to perform a first task in a first stage and pass the output to another a stage comprised of orange cores. *Id.* The green cores operate collectively to perform a second task in a second stage and pass the output to one of a set of stages comprised of orange cores. *Id.* The orange cores perform a third function and serve as the fourth, fifth and sixth stages, although those stages could be considered a single stage for certain purposes and in certain implementations. *Id.*



LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

102. In the Azure-implemented Project Brainwave, an arbitration controller or "network" manages data movement among the memory components: pipeline register files (MRF and VRFs), DRAM, and network I/O queues. Ex. 49 at 6. The arbitration network is deployed on the hardware logic of the FPGA fabric. *Id*. *passim*. This reduces observed latency by up to 90X. *Id*.

103. In 2018 or 2019, Microsoft began incorporating multiple FPGAs onto each server blade. Ex. 37 at Fig. 1, Ex. 38 at Fig. 1. In this iteration of Catapult II, on each server blade a single CPU is placed "behind" and supports several FGPAs and/or ASICs. This is yet another step in the evolution of Azure away from conventional CPUs and toward more widespread use of the FPGA network to provide an ever-increasing fraction of Azure's functions.

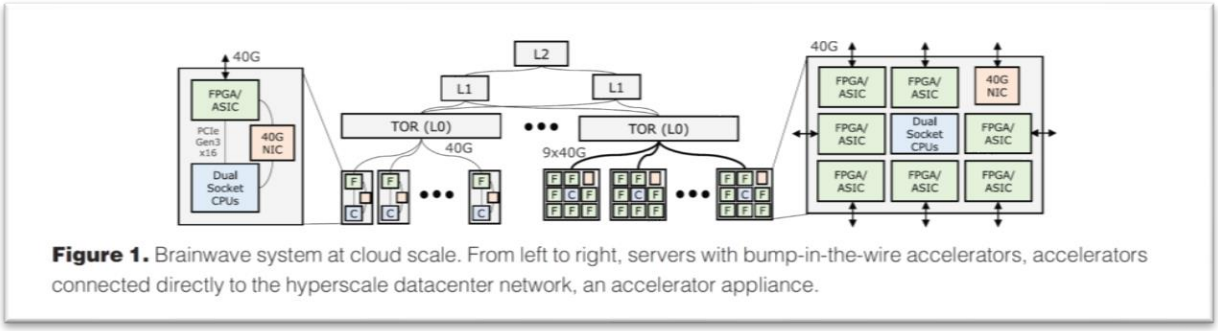


Figure 1. Brainwave system at cloud scale. From left to right, servers with bump-in-the-wire accelerators, accelerators connected directly to the hyperscale datacenter network, an accelerator appliance.

104. In August 2021, several months after the filing of the First Action, one of the lead Azure engineers, Derek Chiou, published a paper describing a Persistent Recurring Neural Network (RNN) deployed on the FPGA-enabled Azure network. Ex. 23 at 6, 23. Dr. Chiou is an author on most of the whitepapers concerning Azure, including Exs. 10-13, 14, 16, 18, and 36. On information and belief, Microsoft is implementing the technology described in Exhibit 23 on at least some Azure servers.

105. RNNs are used for tasks such as speech recognition, text processing and translation. *Id*. Persistent RNNs can be considered a specialized type of DNN in which the inter-stage

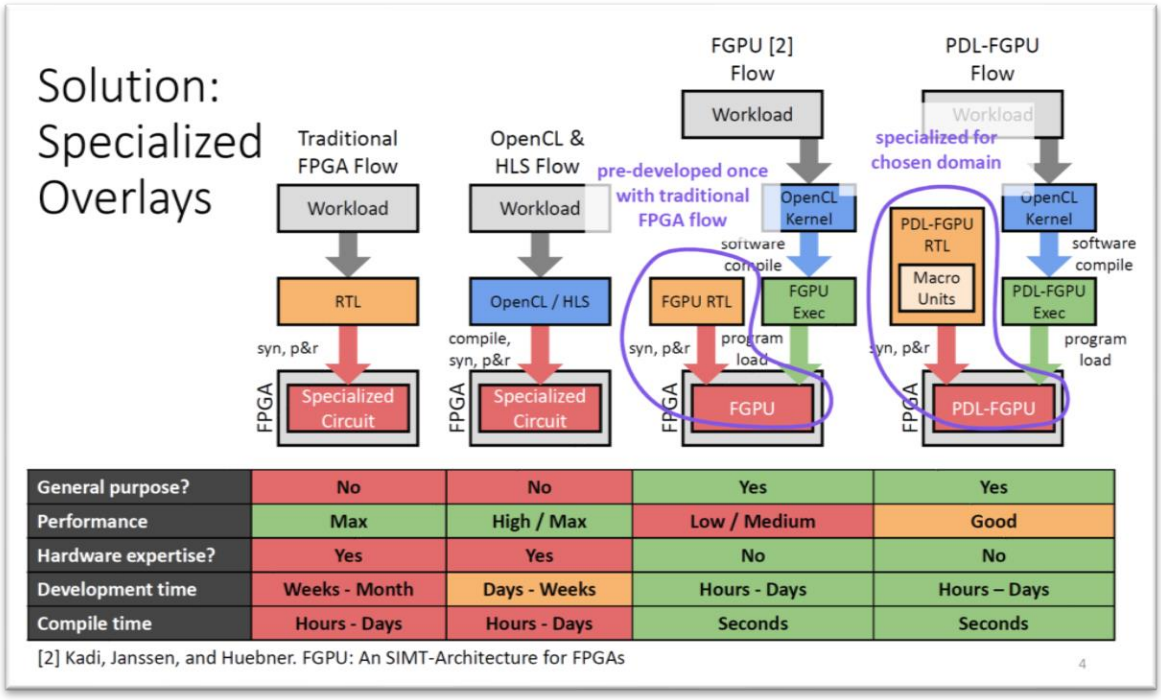
LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

connections are not static but rather change over time and operations may flow upstream, or backwards through the stages. The latter means that information persists because it can travel backwards through the stages and affect the AI model present at an ostensibly upstream node that is also processing data that was received later in time.

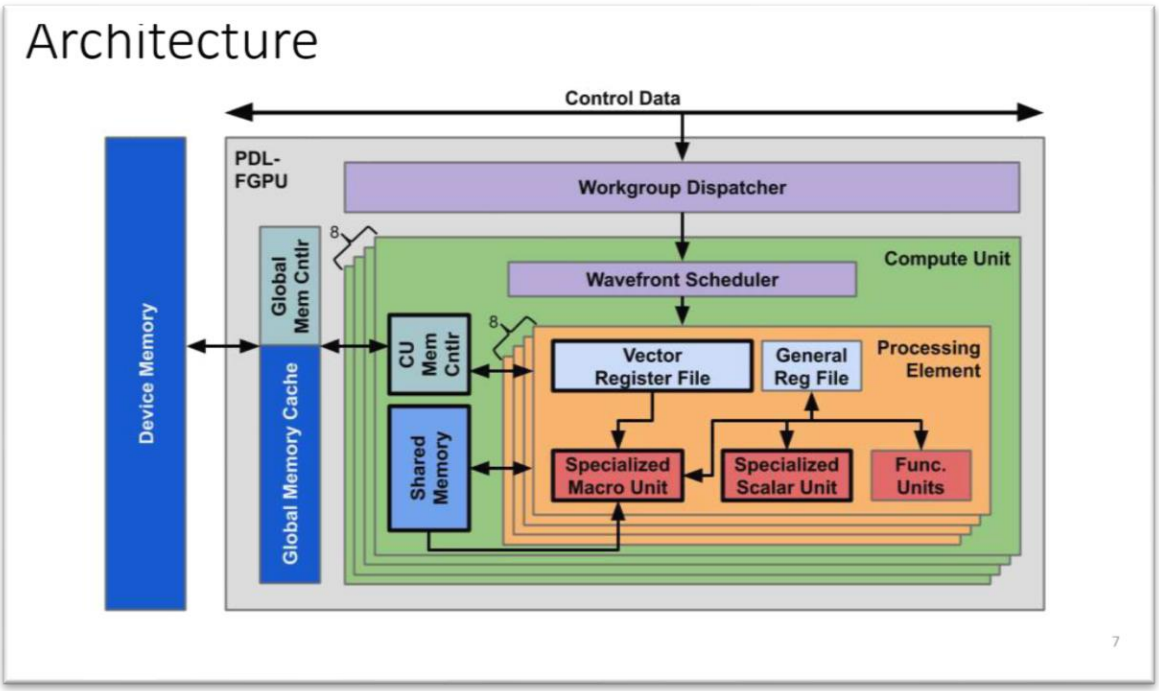
106. In its new implementation of the persistent RNN, Microsoft is using essentially a modified shell for the FGPA[3]. Microsoft's new RNN uses a graphical processing unit (GPU) instantiated on the FPGA. For this reason, it is referred to as an *FPGA general purpose Graphical Processing Unit*, or FGPU. Ex. 23 at 4, Ex. 44. Because it is used to perform persistent deep learning (PDL), it is termed a PDL-FGPU. On information and belief, Microsoft currently deploys PDL-FGPUs on some servers to provide Persistent RNNs on Catapult II.

[3] Recall that shells define the functionality that the FPGA's programmable network of multiplexer-switched ALUs are to perform.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301



107. The PDL-FGPU architecture or shell is shown below. As illustrated, the shell controls memory access, including both SRAM on each FPGA and DRAM on each server blade, and communication between each Processing Element (core) in the Compute Unit.



LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

108. On information and belief, the PDL-FGPU architecture or shell is deployed in conjunction with the Catapult II Shell Components. The PDL-FGPU rides "on top" of the Catapult II Shell Components, thereby allowing the Catapult II Shell-Components to mediate inter-FPGA communication and memory access. On information and belief, Microsoft currently uses this approach on at least some servers.

109. Deploying the RNNs on the FPGA-enabled Azure network via PDL-FGPUs provides a one to three order of magnitude improvement in throughput over Microsoft's internal baseline, which on information and belief is an implementation on a CPU network. See Ex. 23 pp. 13-17. On information and belief, Microsoft currently deploys PDL-FGPUs on at least some FPGA-enabled Azure network servers to provide Persistent RNNs.

110. To summarize, since the introduction of Catapult II in 2015, Microsoft has transitioned a broad swath of workloads and the associated management responsibility from CPUs over to the low-latency FPGA network fabric of Catapult II, per ThroughPuter's patents. In its current form, Catapult II functions as stand-alone, almost entirely independent FPGA fabric that handles the majority of the workload in each Azure datacenter. That independent FPGA network is one of the innovations described in ThroughPuter's patents that has been deliberately copied by Microsoft, including by filing patent applications wrongly claiming these advances as having been invented by Microsoft. In fact, they were disclosed to Microsoft by ThroughPuter years before Microsoft began filing its related patent applications.

111. However, Microsoft was right about one thing – this new FPGA fabric architecture is a "major advance" that has a "profound" "impact [on] the types of workloads that can be accelerated and the scalability of the [system]." Ex. 15 at 54. This design improvement goes "far

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

beyond just an improved network design" and "can be seen as a fundamental shift in the role of CPUs in the datacenter." *Id.*

112. By using the new FPGA-network of Catapult II, as of 2017 Microsoft achieved up to a 150-200 fold improvement in data processing throughput and up to a 50 fold improvement in energy efficiency. Ex. 10 at 34 and exhibits cited therein. Also by that time aggregate latency had been lowered by about a factor of 10. *Id.* On information and belief, in the ensuing four years the evolutions described above (incorporating yet additional functionality described in the ThroughPuter patents) have improved each of these performance metrics by at least another order of magnitude.

113. It's no wonder that Microsoft's CEO described the new FPGA reconfigurable fabric as "magic." *Id.* at 4 and exhibits cited therein.

MICROSOFT'S KNOWLEDGE OF THROUGHPUTER AND ITS PATENTS

114. Starting in 2011, ThroughPuter has invested in development and patent protection of its intellectual property consistently, in order to protect the competitive advantage to which it is entitled under the law as the inventor of the technology at issue in this case. Recognizing the promise of its technology and the benefits it could provide to Microsoft, ThroughPuter made consistent overtures to Microsoft in an effort to collaborate.

115. For example, as discussed above, starting in 2013, Mr. Sandstrom corresponded with Microsoft's Director of Client and Cloud Applications (Dr. Burger: a named inventor on Microsoft's U.S. Patent No. 11,819,657, which is discussed further in Count I below) and others on Microsoft's cloud computing team concerning a potential collaboration between ThroughPuter and Microsoft. *See* Exhibit 19.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

116. At least through that correspondence, and no later than February 2013, Microsoft's cloud computing team was apprised of the technical details of ThroughPuter's Dynamic Parallel Execution Environment™ (DPEE).

117. At least through that correspondence, and no later than February 2013, Mr. Sandstrom provided Microsoft's cloud computing team a copy of the DPEE system description that corresponds with the preferred embodiments disclosed in ThroughPuter's patents. *See* Exhibit 20.

118. From 2013 through 2016, ThroughPuter's President Mr. Sandstrom continued to correspond with Microsoft's cloud computing team concerning a potential collaboration between Microsoft and ThroughPuter. *See, e.g.*, Exhibit 32.

119. For example, in correspondence dated May 12, 2015, in response to a communication from ThroughPuter advising Microsoft its patent portfolio, which at that time included 25 granted US and UK patents, Dr. Burger stated that he had "forwarded [ThroughPuter's] note onto the relevant people looking at any IP and they will contact you directly if interested. Doug." Exhibit 30.

120. Shortly thereafter Microsoft filed a patent application directed to the same subject matter as ThroughPuter's technology naming Dr. Burger, to whom Mr. Sandstrom disclosed ThroughPuter's technology, as an inventor. Following the filing of that application, the leader of Microsoft's cloud computing team declined ThroughPuter's offers for collaboration, saying in an August 3, 2015 email to Mr. Sandstrom: "I don't think we can proceed … we already have our plates very full. Thanks and best wishes." Exhibit 32.

121. Thereafter, Mr. Sandstrom reminded Microsoft several times that the solution he shared was the subject of several U.S. and foreign patents.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

122. In 2015, Mr. Sandstrom had a discussion with Microsoft representative, Dr. Derek Chiou, during which conversation, Dr. Chiou made clear that despite Microsoft's knowledge of ThroughPuter's technology and its patent protection, Microsoft would not be willing to explore any business relationship with ThroughPuter, including with respect to ThroughPuter's intellectual property rights.

123. On May 16, 2016, Mr. Sandstrom informed Arun Justus at Microsoft that "ThroughPuter owns the IP rights for the key techniques that will be necessary in realizing any scalable solution for this must-solve challenge." Exhibit 33.

124. In addition, on May 19, 2016, Mr. Sandstrom wrote Aki Siponen at Microsoft: "[All] cloud service providers will be facing the fundamental scalability solution . . . and ThroughPuter holds the key patented techniques that will be needed in delivering an effective solution to this must-solve challenge." *Id*.

125. On May 17, 2017, via a LinkedIn message, Mr. Sandstrom provided notice to Dr. Burger of the issuance of two ThroughPuter patents. In response to Mr. Sandstrom's message, Dr. Burger, stated "Thanks Mark, appreciate the nice message and will definitely review."

126. In a 2017 Microsoft Research publication titled "The Feniks FPGA Operating System for Cloud Computing," Microsoft describes its experimental development of FPGA processors with a hardware based operating system in highly similar terms as ThroughPuter in the above-referenced, earlier patent disclosures as well as publications such as the one presented by Mr. Sandstrom at FPGAworld in 2014. *Compare e.g.*, Exhibit 43 *with* Exhibit 25.

127. In that 2017 publication, Dr. Chiou is credited with the "initial exploration which provides valuable experience for the system design." (Exhibit 25 at p. 7, Acknowledgement). Dr.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

Chiou is the same individual with whom ThroughPuter spoke in 2015 after Microsoft had acknowledged ThroughPuter's technology and its patent protection.

128. On January 2, 2018, Mr. Sandstrom sent a letter (the "January 2018 letter") to Kevin Scott, Chief Technology Officer of Microsoft Corporation, informing Microsoft of the existence of ThroughPuter's patent portfolio and invited a business discussion.

129. On September 1, 2018, outside patent counsel for ThroughPuter sent Microsoft Chairman John W. Thompson a letter ("the September 2018 letter") offering Microsoft the opportunity to license or acquire "ThroughPuter's Dynamic Parallel Execution (DPE) technology, protected by a portfolio of forty-eight (48) patents issued and pending worldwide ….". The September 2018 letter further explained that to the extent Microsoft's current or planned products were built on certain cloud and enterprise computing technologies, "Microsoft needs to obtain a license from ThroughPuter for continued as well as planned future usage of [ThroughPuter's] patents ….".

130. On September 1, 2018, ThroughPuter's outside patent counsel sent the same letter referred to in the preceding paragraph to Microsoft's then-General Counsel.

131. Microsoft did not respond to any of the letters referred to in the four preceding paragraphs.

132. On information and belief, Microsoft cut off communication with ThroughPuter in or around May 2017 because Microsoft realized that ThroughPuter had been issued patents with earlier priority dates on the technology approach underlying the Microsoft Azure PaaS and its development plans.

133. On information and belief, Dr. Burger and Microsoft's other personnel were instructed to stop communicating with ThroughPuter because Microsoft recognized the similarities

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

between ThroughPuter's patent protected technology and the execution layer of the Azure PaaS and its development plans.

134. Rather than respect ThroughPuter's intellectual property rights, Microsoft made, upon information and belief, the deliberate decision to infringe ThroughPuter's patents. This conscious decision prevents ThroughPuter from being able to compete in the cloud computing space all while Microsoft used the enabling ThroughPuter technology to scale up the world's largest cloud-computing platform. ThroughPuter's patented technology allows Microsoft to claim up to 200-fold technical performance gains.

135. Microsoft recognized the novelty of ThroughPuter's technology by attempting to claim ThroughPuter's technology as its own. Specifically, as discussed herein, Microsoft sought and obtained patent protection on the fundamental technologies invented by ThroughPuter years earlier.

136. Upon information and belief, to date, Microsoft has not disclosed any information concerning ThroughPuter or its technologies to the USPTO in connection with any Microsoft patent application related to Azure.

137. On information and belief, Microsoft has been monitoring ThroughPuter's patent portfolio since at least 2017 and has had actual notice of all ThroughPuter patents that issued since that time.

138. In a recently filed petition for *inter partes* review directed to the '242 patent, Microsoft admitted that it has been monitoring ThroughPuter's patent family and has knowledge of the claims asserted herein, which claim priority through the patent application that matured into the '242 patent. Such monitoring provided Microsoft with actual notice of the two patents asserted herein.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

139. On information and belief, Microsoft has been monitoring ThroughPuter's patent portfolio since at least the time it was served with the complaint in the Virginia Action.

140. Despite this knowledge, Microsoft has made the deliberate choice to refuse any business arrangement or resolution with ThroughPuter, opting instead to infringe ThroughPuter's patents, thereby preventing ThroughPuter from competing in the market ThroughPuter's technology has enabled.

141. On information and belief, Microsoft's infringement has been continuous, deliberate and in willful disregard of ThroughPuter's patent rights.

COUNT 1

(Infringement of U.S. Patent No. 11,150,948)

142. ThroughPuter repeats and realleges each and every allegation contained above as though fully set forth herein.

143. On October 19, 2021, the United States Patent and Trademark Office duly and legally issued the '948 patent, entitled "Managing Programmable Logic-Based Processing Unit Allocation on a Parallel Data Processing Platform." A copy of the '948 patent is attached as Exhibit 8.

144. Mark Sandstrom is the sole and true inventor of the '948 patent.

145. ThroughPuter, Inc. owns all right, title, and interest to and in the '948 patent.

146. Microsoft infringes claims 1-15 of the '948 patent.

147. As demonstrated in the side-by-side comparison below, claim 1 of ThroughPuter's '948 patent closely matches the claims of Microsoft's '657 patent. Ex. 42. In the chart below, the column on the left shows claim 1 of the '948 patent, which claims priority to applications dating back to 2012. The column on the right shows independent claim 8 of Microsoft's '657 patent,

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

which claims priority to an application filed in 2015. As can be appreciated from this side-by-side comparison, Microsoft obtained a patent on substantially the same technology taught by ThroughPuter's patent applications. However, ThroughPuter's '948 patent is entitled to a priority date that is over two years earlier than Microsoft's '657 patent.

ThroughPuter's U.S. Patent No. 11,150,948 Independent Claim 1	Microsoft's U.S. Patent No. 10,819,657, Independent Claim 8
A method on a parallel data processing platform comprising a plurality of programmable logic-based processing units for accelerating a service comprising at least a first program and a second program, different than the first program, the method comprising:	A method in a data center comprising a pool of acceleration components for accelerating the service comprising at least a first service functionality and a second service functionality, different from the first functionality, the method comprising:
forming a first set of inter-task communication paths connecting a first set of programmable logic-based processing units of the plurality of programmable logic-based processing units into a first multi-stage program instance, wherein the first multi-stage program instance is configured to accelerate the first program, and wherein the first set of programmable logic-based processing units are interconnected using reconfigurable cross-connects;	forming a first graph comprising a first set of interconnected acceleration components, wherein the first graph is configured to accelerate the first service functionality corresponding to the service, and wherein each of the first set of acceleration components comprises a programmable logic array comprising hardware logic blocks interconnected using reconfigurable interconnects;
forming a second set of inter-task communication paths connecting a second set of programmable logic-based processing units of the plurality of programmable logic-based processing units to a second multi-stage program instance, wherein the second multi-stage program instance is configured to accelerate the second program, and wherein the second set of programmable logic-based processing units are interconnected using reconfigurable cross-connects;	forming a second graph comprising a second set of interconnected acceleration components, wherein the second graph is configured to accelerate the second service functionality corresponding to the service, and wherein each of the second set of acceleration components comprises a programmable logic array comprising hardware logic blocks interconnected using reconfigurable interconnects;
in response to a first increased demand for the first multi-stage program instance determined by monitoring a first characteristic corresponding to the first set of programmable logic-based processing units, based on an adaptive optimized resource allocation policy, adding at least one more programmable logic-based processing unit to the first set of programmable logic-based processing units from the plurality of programmable logic-based processing units; and	in response to a first increased demand for hardware acceleration from the service determined by monitoring a first characteristic corresponding to the first set of interconnected acceleration components, based on an allocation policy, a service management component adding at least one more acceleration component to the first set of interconnected acceleration components from the pool of acceleration components;

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

in response to a second increased demand for the second multi-stage program instance determined by monitoring a second characteristic different from the first characteristic, corresponding to the second set of programmable logic-based processing units, based on the adaptive optimized resource allocation policy, adding at least one more programmable logic-based processing unit to the second set of programmable logic-based processing units from the plurality of programmable logic-based processing units.	and in response to a second increased demand for hardware acceleration from the service determined by monitoring a second characteristic, different from the first characteristic, corresponding to the second set of interconnected acceleration components, based on the allocation policy, the service management component adding at least one more acceleration component to the second set of interconnected acceleration components from the pool of acceleration components.
based on at least the determination, the given target programmable logic core redirects the request to another programmable logic core implementing the designated task.	based on at least the determination, the target hardware acceleration device redirects the request to another hardware acceleration device implementing the designated hardware accelerated service.
configuring the selected field-programmable gate arrays to process a respective processing stage of a respective requesting instance, and	configuring the selected field-programmable gate arrays of the plurality to perform the individual stages of the different functions; and
configuring certain selected field-programmable gate arrays to support communicating, by the task executing on the respective field-programmable gate array, final results to a requesting client over a network in the data processing system.	configuring certain selected field-programmable arrays to communicate final results of the different functions to the requesting server unit components, the certain selected field-programmable arrays communicating the final results to the requesting server unit components over a network in the data processing system.

148. On information and belief, Microsoft believed at the time of filing and still believes that the subject matter claimed in the ’657 patent is patent eligible under 35 U.S.C. § 101.

149. On information and belief, Microsoft believed at the time of filing and still believes that the subject matter claimed in the ’657 patent is novel under 35 U.S.C. § 102 in view of the references disclosed to and considered by the United States Patent & Trademark Office (“USPTO”) during examination of the underlying application.

150. On information and belief, Microsoft believed at the time of filing and still believes that that the subject matter claimed in the ’657 patent is non-obvious under 35 U.S.C. § 103 in view of the references Microsoft disclosed to and considered by the USPTO during examination

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

of the underlying application.

151. Claim 1 of the ’948 patent is representative of the claims infringed by Microsoft and recites:

> 1. A method on a parallel data processing platform comprising a plurality of programmable logic-based processing units for accelerating a service comprising at least a first program and a second program, different than the first program, the method comprising:
>
> forming a first set of inter-task communication paths connecting a first set of programmable logic-based processing units of the plurality of programmable logic-based processing units into a first multi-stage program instance, wherein the first multi-stage program instance is configured to accelerate the first program, and wherein the first set of programmable logic-based processing units are interconnected using reconfigurable cross-connects;
>
> forming a second set of inter-task communication paths connecting a second set of programmable logic-based processing units of the plurality of programmable logic-based processing units to a second multi-stage program instance, wherein the second multi-stage program instance is configured to accelerate the second program, and wherein the second set of programmable logic-based processing units are interconnected using reconfigurable cross-connects;
>
> in response to a first increased demand for the first multi-stage program instance determined by monitoring a first characteristic corresponding to the first set of programmable logic-based processing units, based on an adaptive optimized resource allocation policy, adding at least one more programmable logic-based processing unit to the first set of programmable logic-based processing units from the plurality of programmable logic-based processing units; and
>
> in response to a second increased demand for the second multi-stage program instance determined by monitoring a second characteristic different from the first characteristic, corresponding to the second set of programmable logic-based processing units, based on the adaptive optimized resource allocation policy, adding at least one more programmable logic-based processing unit to the second set of programmable logic-based processing units from the plurality of programmable logic-based processing units.

152. On information and belief, Azure is implemented in a manner that meets each and every limitation of claim 1 of the ’948 patent.

153. On information and belief, Azure practices the invention recited in claim 8 of Microsoft’s ‘657 patent.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

1. A method on a parallel data processing platform comprising a plurality of programmable logic-based processing units for accelerating a service comprising at least a first program and a second program, different than the first program, the method comprising:

154. Azure comprises a parallel data processing system with a plurality of programmable logic-based processing units (FPGA processors) for accelerating multiple applications such as DNNs, RNNs, compression, web search ranking, speech recognition, and SDN. In some applications a single application or service (e.g., speech recognition) may have multiple implementations (e.g., English and German) associated with it. In Azure, applications or services are often broken down into smaller functional units that are executed (accelerated) separately.

forming a first set of inter-task communication paths connecting a first set of programmable logic-based processing units of the plurality of programmable logic-based processing units into a first multi-stage program instance, wherein the first multi-stage program instance is conf5igured to accelerate the first program, and wherein the first set of programmable logic-based processing units are interconnected using reconfigurable cross-connects;

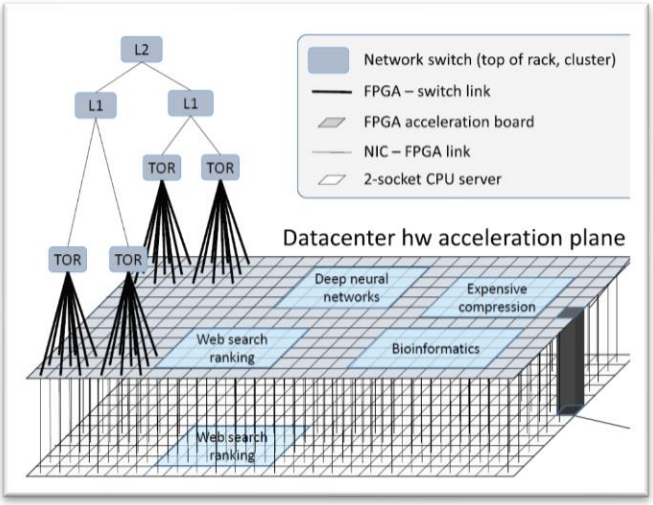
155. Microsoft uses Azure to form a first set of inter-task communication paths connecting a first set of programmable logic-based processing units of the plurality of programmable logic-based processing units into a first multi-stage program instance.

156. Azure includes multiple FPGAs, which are programmable logic based processing units. Microsoft connects a first set of FPGAs into a multi-stage program instance comprised of multiple FPGA cores. Every FPGA in an Azure datacenter is able to communicate with every other FPGA in a datacenter through a high-speed, low latency, multiplexer switched fabric. Exs. 10, 14, 36 *passim*.

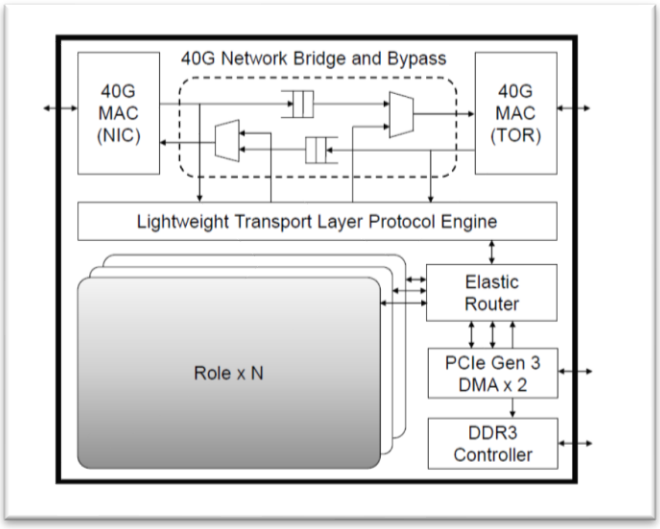
157. Each Azure rack in a datacenter is connected by a two-tier switching network L1/L2 as shown below. Exs. 10, 14, *passim*. The packet communications within each rack and to the top-

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

of-rack switch (TOR) are controlled by the FPGA via the high-speed, low-latency multiplexer controlled connections that serve as the Catapult II Shell Components. *Id.*

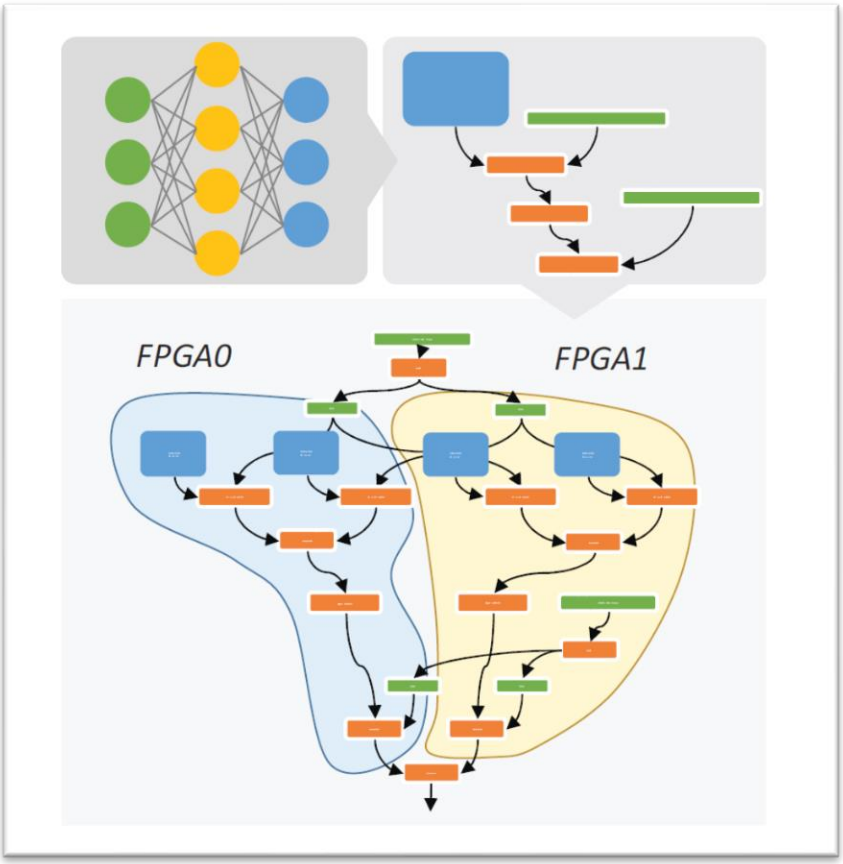


158. The hardware through which the FPGAs communicate constitutes a first set of inter-task communication paths connecting a first set of programmable logic-based processing units. The figure below shows the "shell" of the Catapult II FPGA. Exs. 10, 14. All of the depicted functionality is performed on the FPGA itself, which is essentially an array of multiplexers and wires that provide high-speed, low latency interconnectivity between logical units. *Id.*, *see also* Ex. 46 *passim*, Ex. 16 at 5.



LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

159. The FPGA cores for a given program flow can be spread among multiple FPGA stages. Ex. 18 at 10-12, Ex. 37 at 21-22. The group of FPGA cores that are performing the same task constitute a stage. *Id.* In the illustration below (an excerpt of Ex. 18 Fig. 3), the tasks by the DNN model are separated into three types of tasks, illustrated in blue, orange and green. *Id.* Each of tasks is deployed on a FPGA processing core instantiated on one of two FPGA processing stages, labelled FPGA0 and FPGA1. *Id.* The blue cores operate collectively to perform a first task in a first stage and pass the output to another a stage comprised of orange cores. *Id.* The green cores operate collectively to perform a second task in a second stage and pass the output to one of a set of stages comprised of orange cores. *Id.* The orange cores perform a third function and serve as the fourth, fifth and sixth stages, although those stages could be considered a single stage for certain purposes and in certain implementations. *Id.*



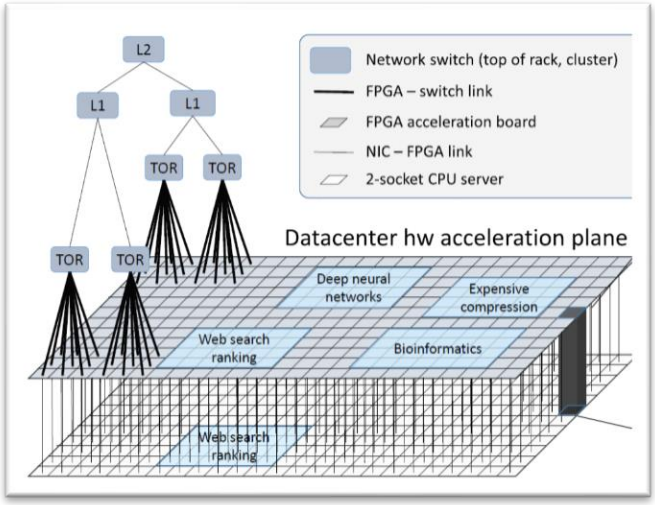
LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

forming a second set of inter-task communication paths connecting a second set of programmable logic-based processing units of the plurality of programmable logic-based processing units to a second multi-stage program instance, wherein the second multi-stage program instance is configured to accelerate the second program, and wherein the second set of programmable logic-based processing units are interconnected using reconfigurable cross-connects;

160. Microsoft uses Azure to form a second set of inter-task communication paths connecting a second set of programmable logic-based processing units of the plurality of programmable logic-based processing units into a first multi-stage program instance.

161. Azure includes multiple FPGA processors, which are programmable logic based processing units. Microsoft connects a second set of FPGA processors to form a multi-stage program instance comprised of multiple FPGA cores. Every FPGA in an Azure datacenter is able to communicate with every other FPGA in a datacenter through a high-speed, low latency, multiplexer switched fabric. Exs. 10, 14, 36 *passim*.

162. Each Azure rack in a datacenter is connected by a two-tier switching network L1/L2 as shown below. Exs. 10, 14, *passim*. The packet communications within each rack and to the top-of-rack switch (TOR) are controlled by the FPGA via the high-speed, low-latency multiplexer controlled connections that serve as the Catapult II Shell Components. *Id.*

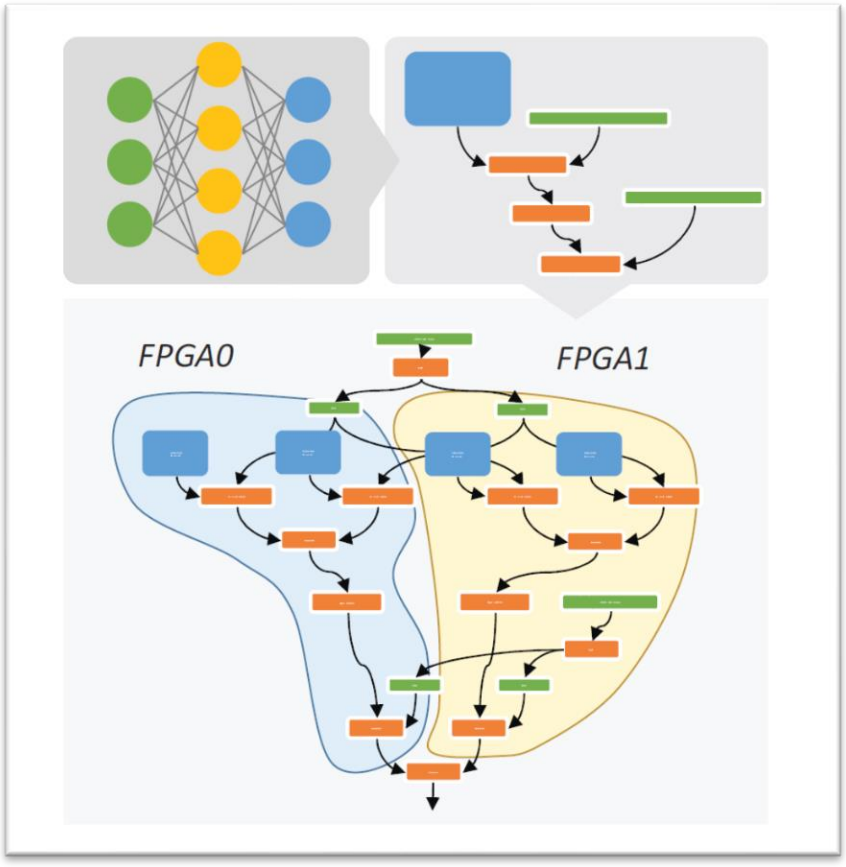


LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

163. The hardware through which the FPGAs communicate constitutes a first set of inter-task communication paths connecting a first set of programmable logic-based processing units.

164. The FPGA cores for a given program flow can be spread among multiple FPGA stages. Ex. 18 at 10-12, Ex. 37 at 21-22. The group of FPGA cores that are performing the same task constitute a stage. *Id.* In the illustration below (an excerpt of Ex. 18 Fig. 3), the tasks by the DNN model are separated into three types of tasks, illustrated in blue, orange and green. *Id.* Each of tasks is deployed on a FPGA processing core instantiated on one of two FPGA processing stages, labelled FPGA0 and FPGA1. *Id.* The blue cores operate collectively to perform a first task in a first stage and pass the output to another a stage comprised of orange cores. *Id.* The green cores operate collectively to perform a second task in a second stage and pass the output to one of a set of stages comprised of orange cores. *Id.* The orange cores perform a third function and serve as the fourth, fifth and sixth stages, although those stages could be considered a single stage for certain purposes and in certain implementations. *Id.*

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301



165. Azure simultaneously accelerates multiple multi-stage services or applications. At any given point in time, an Azure datacenter is accelerating multiple DNNs, RNNs, web search ranking, speech recognition, compression, computer vision, speech translation, and other programs. Many of these program or service instances are multi-stage.

> ***in response to a first increased demand for the first multi-stage program instance determined by monitoring a first characteristic corresponding to the first set of programmable logic-based processing units, based on an adaptive optimized resource allocation policy, adding at least one more programmable logic-based processing unit to the first set of programmable logic-based processing units from the plurality of programmable logic-based processing units; and***

166. In response to a first increased demand for the first multi-stage program instance determined by monitoring a first characteristic corresponding to the first set of programmable logic-based processing units, based on an adaptive optimized resource allocation policy, Azure

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

adds at least one more programmable logic-based processing unit to the first set of programmable logic-based processing units from the plurality of programmable logic-based processing units.

167. As Microsoft explained in a white paper discussing the Catapult II architecture, "[a]s demand for a service grows or shrinks, a global manager grows or shrinks the pools correspondingly." Ex. 10 at 2, see also Ex. 14 at 4. The pools of cores among and between different FPGAs assigned to each multi-stage program instance is controlled by the Catapult II Shell Components under the supervision and control of the Service Manager and Resource Manager. Ex. 10 at 10. As discussed above, load data, which is a first characteristic as claimed, is monitored and tracked at each FPGA core group. See Ex.41. That data is published to other FPGA core groups in the datacenter. Based on this monitoring, Azure expands and contracts the number of cores assigned to each group over time.

in response to a second increased demand for the second multi-stage program instance determined by monitoring a second characteristic different from the first characteristic, corresponding to the second set of programmable logic-based processing units, based on the adaptive optimized resource allocation policy, adding at least one more programmable logic-based processing unit to the second set of programmable logic-based processing units from the plurality of programmable logic-based processing units.

168. In response to a second increased demand for the second multi-stage program instance determined by monitoring a second characteristic different from the first characteristic, corresponding to the second set of programmable logic-based processing units, based on the adaptive optimized resource allocation policy, Azure adds at least one more programmable logic-based processing unit to the second set of programmable logic-based processing units from the plurality of programmable logic-based processing units.

169. Multiple instances of such multi-stage programs are executed in similar manner described above in connection with the first multi-stage program instance. That discussion is incorporated herein by reference.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

170. As described in Microsoft's own patent filings, "the load data [need] not [be] stored as a single discrete value. For example, the load data 414 may include a plurality of data types and values. Such as, for example, a number of queued requests for the target hardware acceleration device, a recent processing time of a previous request, an estimate based on the number and type of requests in the queue, a total size of the queued requests, a round trip time for receiving responses to requests, and any other suitable types of data that may indicate a load of the target hardware acceleration device." Ex.41, 472 Patent at 8:22-24. Each of the foregoing represents a second characteristic different than the first characteristic as claimed.

171. On information and belief, Azure uses a plurality of these different data types to reallocate cores among the multi-stage program instances in a given data center on any given day. Any two such multi-stage program instances being rebalanced or reallocated according to given load data types may constitute the first and second multi-stage program instances.

172. Upon information and belief, in accordance with 35 U.S.C. § 287, Microsoft has had actual notice and knowledge of the '948 patent no later than its issuance.

173. Microsoft continues without license to make, use, import, offer for sale, and/or sell in the United States services or products that infringe the '948 patent including specifically Microsoft Azure and its cloud computing functionalities.

174. Microsoft has directly and indirectly infringed and continues to directly and indirectly infringe the '948 patent by engaging in acts constituting infringement under 35 U.S.C. § 271(a) including but not necessarily limited to one or more of making, using, selling and offering to sell, in this District and elsewhere in the United States, and importing into the United States, the Microsoft Azure platform or components and services thereof.

175. Microsoft's continued infringement of the '948 patent is knowing, intentional, and

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

willful.

176. Microsoft has had knowledge of and notice of the chain of applications underlying the '948 patent since at least May 2015, when Microsoft representatives acknowledged receipt of an email from ThroughPuter disclosing several of ThroughPuter's U.S. patent applications and in any case no later than January 2018 when ThroughPuter brought ThroughPuter's patent portfolio to the attention of Microsoft in the January 2018 letter, and despite this knowledge continues to commit the aforementioned infringing acts. For at least the reasons stated in this paragraph and above, this infringement has been willful.

177. Microsoft has been actively monitoring ThroughPuter's patent portfolio since at least the time it was served with the complaint in the Virginia Action and has gained actual knowledge of the patents asserted herein.

178. Microsoft actively, knowingly, and intentionally has induced, or has threatened to induce, infringement of the '948 patent through a range of activities.

179. First, on information and belief, Microsoft has induced infringement by, with knowledge of the '948 patent, controlling the design and development of, offering for sale, and selling the services of the Azure platform with the knowledge and specific intent that its customers will use the Azure platform to infringe the '948 patent by executing the system operations and utilizing the system components to perform dynamic resource management of the pool of Azure processing resources on behalf of application programs of customers of the Azure platform cloud computing services.

180. Second, on information and belief, Microsoft has, with knowledge of the '948 patent, induced infringement by its customers through the dissemination of promotional, marketing, and tutorial materials relating to the Azure platform with the knowledge and specific

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

intent that its customers will use the Azure platform to infringe the '948 patent by executing the system operations and utilizing the system components to perform dynamic resource management of the pool of Azure processing resources on behalf of application programs of customers of the Azure platform cloud computing services.

181. Third, on information and belief, Microsoft has, with knowledge of the '948 patent, induced infringement by its customers through the creation and online posting of tutorial and "how-to" materials for the Azure platform and/or its individual components in the United States with the knowledge and specific intent that its customers will use the Azure platform to infringe the '948 patent by executing the system operations and utilizing the system components to perform dynamic resource management of the pool of Azure processing resources on behalf of application programs of customers of the Azure platform cloud computing services.

182. Fourth, on information and belief, Microsoft has, with knowledge of the '948 patent, induced infringement through the distribution of other instructional materials, product manuals, and technical materials with the knowledge and the specific intent to encourage and facilitate its customers' infringing use of the Azure platform.

183. Microsoft has engaged in the above activities with knowledge of the '948 patent and with the specific intent to encourage and cause infringement by its customers, as shown by the allegations set forth above.

184. Microsoft has contributed to, or has threatened to contribute to, the infringement by its customers of the '948 patent by, without authority, selling and offering to sell within the United States cloud computing services and customer support services for practicing the claimed invention of the '948 patent, including at least the Azure platform as a whole and/or the individual components of the Azure platform. When, for example, the Azure platform is used by Microsoft's

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

customers for the various cloud computing services Microsoft offers, the Azure system operations and system components are used to perform the claimed dynamic resource management of the pool of Azure processing resources on behalf of application programs of customers of the Azure cloud platform, thereby infringing the '948 patent. In such settings, the Azure platform and/or its individual components, supplied by Microsoft, constitute at least a material part of the claimed invention of the '948 patent. Microsoft's infringement of the '948 patent has injured ThroughPuter in its business and property rights.

185. For example, Microsoft supplies certain customers with an Azure Stack Edge Pro FPGA, which provides customers with a Hardware-as-a-service solution. *See* https://docs.microsoft.com/en-us/azure/databox-online/azure-stack-edge-overview. The Azure Stack Edge Pro FPGA can be used for "rapid Machine Learning (ML) inferencing at the edge and preprocessing data before sending it to Azure."

186. The Azure Stack Edge Pro FPGA is a component of a patented system that is used in practicing a patent process that constitutes at least a material part of the invention. The Azure Stack Edge Pro FPGA had no substantial non-infringing use at least because any non-infringing use for machine learning would be *de minimis*.

187. Microsoft's infringement of the '948 patent has been and is deliberate and willful and constitutes egregious misconduct. On information and belief, despite actual knowledge of the '948 patent and numerous related patents and applications since at least May 2015, Microsoft continued to develop and offer its infringing products and services. In developing and offering its products and services, Microsoft has been willfully blind to this ongoing infringement.

188. Pursuant to 35 U.S.C. § 284, ThroughPuter is entitled to recover monetary damages for the injuries arising from Microsoft's willful infringement in an amount to be determined at

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

trial. Microsoft's infringement of the '948 patent has caused irreparable harm to ThroughPuter and will continue to cause such harm unless and until Microsoft's infringing activities are enjoined by this Court.

189. Microsoft's infringement of the '948 patent is exceptional and entitles ThroughPuter to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

COUNT 2

(Infringement of U.S. Patent No. 11,036,556)

190. ThroughPuter repeats and realleges each and every allegation contained above as though fully set forth herein.

191. On June 15, 2021, the United States Patent and Trademark Office duly and legally issued the '556 patent, entitled "Concurrent Program Execution Optimization." A copy of the '556 patent is attached as Exhibit 1.

192. Mark Sandstrom is the sole and true inventor of the '556 patent.

193. ThroughPuter, Inc. owns all right, title and interest to and in the '556 patent.

194. Microsoft infringes claims 1-8 of the '556 patent.

195. As demonstrated in side-by-side comparison, claim 1 of ThroughPuter's '556 patent closely matches claim 17 of Microsoft's '392 patent. Ex. 47. The column on the left shows claim 1 of the '556 patent, which claims priority to applications dating back to 2012. The column on the right shows independent claim 17 of Microsoft's '392 patent, which claims priority to an application filed in 2015. As can be appreciated from this side-by-side comparison, Microsoft obtained a patent on substantially the same technology taught by ThroughPuter's patent

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

applications. However, ThroughPuter's '556 patent is entitled to a priority date that is more than two years earlier than Microsoft's.

ThroughPuter's U.S. Patent No. 11,556,556, Independent Claim 1	Microsoft's U.S. Patent No. 10,296,392, Independent Claim 17
A method performed in a data processing system, the method comprising:	A method performed in a data processing system, the method comprising:
receiving, by hardware logic and/or software logic, requests to perform different tasks on behalf of instances of a plurality of programs managed by a data processing system;	receiving requests to perform different functions on behalf of tenant functionality that executes on requesting server unit components of the data processing system;
identifying, by the hardware logic and/or software logic for each of the instances, communication interdependencies between different processing stages of a set of processing stages of the respective instance;	parsing the different functions to identify programmatic calls between different stages of the different functions;
based on conditions in the data processing system, dynamically varying, by the hardware logic and/or software logic, structures of field-programmable gate arrays used to process different tasks of the instances of the plurality of programs, the structures being dynamically varied by	based at least on conditions in the data processing system, dynamically varying structures of field-programmable gate arrays used to implement different invocations of the different functions, the structures being dynamically varied by:
identifying available field-programmable gate arrays of the data processing system that are available to process different processing stages of requesting instances of respective programs,	identifying available field-programmable arrays of the data processing system that are available to implement the different stages of the different functions;
based at least on the conditions in the data processing system, identifying selected field-programmable gate arrays from the available field-programmable gate arrays to execute the different processing stages of the requesting instances of the respective programs,	based at least on the conditions in the data processing system, identifying selected field-programmable gate arrays from the available field-programmable gate arrays to perform individual stages of the different functions;
configuring the selected field-programmable gate arrays to process a respective processing stage of a respective requesting instance, and	configuring the selected field-programmable gate arrays of the plurality to perform the individual stages of the different functions; and
configuring certain selected field-programmable gate arrays to support communicating, by the task executing on the respective field-programmable gate array, final results to a requesting client over a network in the data processing system.	configuring certain selected field-programmable arrays to communicate final results of the different functions to the requesting server unit components, the certain selected field-programmable arrays communicating the final results to the requesting server unit components over a network in the data processing system.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

196. On information and belief, Microsoft believed at the time of filing and still believes that the subject matter claimed in the '392 patent is patent eligible under 35 U.S.C. § 101.

197. On information and belief, Microsoft believed at the time of filing and still believes that the subject matter claimed in the '392 patent is novel under 35 U.S.C. § 102 in view of the references disclosed to and considered by the United States Patent & Trademark Office ("USPTO") during examination of the underlying application.

198. On information and belief, Microsoft believed at the time of filing and still believes that that the subject matter claimed in the '392 patent is non-obvious under 35 U.S.C. § 103 in view of the references Microsoft disclosed to and considered by the USPTO during examination of the underlying application.

199. Claim 1 of the '556 patent is representative of the claims infringed by Microsoft and recites:

> 1. A method performed in a data processing system, the method comprising:
>
> receiving, by hardware logic and/or software logic, requests to perform different tasks on behalf of instances of a plurality of programs managed by a data processing system;
>
> identifying, by the hardware logic and/or software logic for each of the instances, communication interdependencies between different processing stages of a set of processing stages of the respective instance;
>
> based on conditions in the data processing system, dynamically varying, by the hardware logic and/or software logic, structures of field-programmable gate arrays used to process different tasks of the instances of the plurality of programs, the structures being dynamically varied by
>
> identifying available field-programmable gate arrays of the data processing system that are available to process different processing stages of requesting instances of respective programs,
>
> based at least on the conditions in the data processing system, identifying selected field-programmable gate arrays from the available field-programmable gate arrays to execute the different processing stages of the requesting instances of the respective programs,

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

configuring the selected field-programmable gate arrays to process a respective processing stage of a respective requesting instance, and

configuring certain selected field-programmable gate arrays to support communicating, by the task executing on the respective field-programmable gate array, final results to a requesting client over a network in the data processing system.

200. On information and belief, Azure is implemented in a manner that meets each and every limitation of claim 1 of the '556 patent.

201. On information and belief, Azure practices the invention recited in claim 17 of Microsoft's '392 patent (Ex. 47).

1. A method performed in a data processing system, the method comprising:

202. Azure is a data processing system.

receiving, by hardware logic and/or software logic, requests to perform different tasks on behalf of instances of a plurality of programs managed by a data processing system;

203. Azure receives by hardware logic/and or software logic, requests to perform different tasks on behalf of instances of a plurality of programs managed by a data processing system.

204. Azure, e.g. via Catapult II Shell Components instantiated on FPGAs, receives requests to accelerate services from at least the CPUs by hardware logic and/or software logic that are positioned "behind" the FPGAs in the Catapult II network. Ex. 15 at 54. On information and belief, in Azure's current form a hardware microservice running on an FPGA may also initiate a request for acceleration.

identifying, by the hardware logic and/or software logic for each of the instances, communication interdependencies between different processing stages of a set of processing stages of the respective instance;

205. Azure identifies, by the hardware logic and/or software logic for each of the instances, communication interdependencies between processing stages of a set of processing

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

stages of the respective instance.

206. In Azure, under the supervision of the RMs the SM components instantiated on the CPUs and/or FPGA shells determine how to break a requested service down into steps or sub-task components. Ex. 35 at 119, Ex. 18 at 11. Those sub-tasks are assigned to various processing cores on the FPGA fabric. As part of this process, the SM components determine what each sub-task will provide as inputs and outputs, which involves identification of communication interdependencies between the stages (e.g. which stage needs to communicate with which other stage).

based on conditions in the data processing system, dynamically varying, by the hardware logic and/or software logic, structures of field-programmable gate arrays used to process different tasks of the instances of the plurality of programs,

207. Based on conditions in the data processing system, Azure dynamically varies, by the hardware logic and/or software logic, structures of field-programmable gate arrays used to process different tasks of the instances of the plurality of programs.

208. Azure's AC, RM, SMs and FMs works in conjunction to assign tasks of a multi-stage program instance to a group of reconfigurable FPGA processors. When a core is reallocated to perform a different task (for instance speech recognition instead of compression), the FPGA circuit is reconfigured accordingly by loading a corresponding image file or role. Ex. 10 at 7, Ex. 35 at 116. Such reconfiguration varies the structure of the logic blocks of the FPGA.

the structures being dynamically varied by identifying available field-programmable gate arrays of the data processing system that are available to process different processing stages of requesting instances of respective programs,

209. In Azure the process of dynamically varying structures of available field-programmable gate arrays of the data processing system, includes identifying FPFAs that are available to process different processing stages of requesting instances of respective programs.

210. As discussed above in connection with previous counts and/or in the section of this

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

complaint entitled Microsoft's Infringing Cloud Computing Architecture, Azure, e.g. via the Catapult II system, monitors load data of various types and reallocates FPGA processing cores accordingly. This reallocation includes identifying available FPGAs. For instance, unused or available cores are allocated to "join" groups that are experiencing particularly high demand.

based at least on the conditions in the data processing system, identifying selected field-programmable gate arrays from the available field-programmable gate arrays to execute the different processing stages of the requesting instances of the respective programs,

211. Azure, based at least on the conditions in the data processing system, identifies selected field-programmable gate arrays from the available field-programmable gate arrays to execute the different processing stages of the requesting instances of the respective programs.

212. As discussed above, after such a reallocation involving a multi-stage stage program instance (e.g. an instance of a DNN service), the SM(s) in cooperation with the FMs determine which sub-tasks will be executed on which cores of which newly allocated FPGA blocks. These assignments are made based in part on the load data reported concerning processing tasks placed on each FPGA or block thereof, which data comprises the conditions in the data processing system.

configuring the selected field-programmable gate arrays to process a respective processing stage of a respective requesting instance, and

213. Azure configures the selected field-programmable gate arrays to process a respective processing stage of a respective.

214. As discussed above, when cores are allocated for execution of a new function that requires execution of a new function, FPGA core blocks may be re-imaged as needed. That re-imaging may be partial (only some cores on the FPGA) or complete (all processing cores on the FPGA). Typically, the Catapult II Shell Components are not re-imaged during this process. Such reimaging constitutes configuring the FPGA gate arrays as claimed.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

configuring certain selected field-programmable gate arrays to support communicating, by the task executing on the respective field-programmable gate array, final results to a requesting client over a network in the data processing system.

215. Azure configures selected field-programmable gate arrays to support communicating, by the task executing on the respective field-programmable gate array, final results to a requesting client over a network in the data processing system.

216. The head component of each group of cores (each instantiated on an FGPA) is configured under the supervision of the RM and the SM components to communicate the final result of the acceleration request to the requesting client, hosted e.g. on the CPU which is local to the FPGA launching the acceleration request. On information and belief, in its current form Azure, e.g. via Catapult II, permits requesting clients to reside on FPGAs (e.g., as hardware accelerated microservices). Azure also allows external CPUs to communicate with FPGAs.

217. Upon information and belief, in accordance with 35 U.S.C. § 287, Microsoft has had actual notice and knowledge of the '556 patent no later than its issuance.

218. Microsoft has been actively monitoring ThroughPuter's patent portfolio since at least the time it was served with the complaint in the Virginia Action and has gained actual knowledge of the patents asserted herein.

219. Microsoft continues, without license, to make use, offer for sale, import and/or sell in the United States services or products that infringe the '556 patent including specifically Microsoft Azure and its cloud computing functionalities.

220. Microsoft has directly infringed and continues to directly infringe the '556 patent by engaging in acts constituting patent infringement under 35 U.S.C. § 271(a) including but not necessarily limited to one or more of making, using, selling and offering to sell, in this District and elsewhere in the United States, and importing into the United States, the Microsoft Azure platform or components and services thereof.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

221. Microsoft's continuing infringement of the '556 patent is knowing, intentional, and willful.

222. Microsoft has had knowledge of and notice of the chain of applications underlying the '556 patent since at least May 2015, when Microsoft representatives acknowledged receipt of an email from ThroughPuter disclosing several of ThroughPuter's U.S. patent applications and in any case no later than January 2018 when ThroughPuter brought ThroughPuter's patent portfolio to the attention of Microsoft in the January 2018 letter, and despite this knowledge continues to commit the aforementioned infringing acts – and despite this knowledge continues to commit the aforementioned infringing acts. For at least the reasons stated in this paragraph and above, this infringement has been willful.

223. Microsoft actively, knowingly, and intentionally has induced, or has threatened to induce, infringement of the '556 patent under 35 U.S.C. § 271(b) and (c) through a range of activities.

224. First, on information and belief, Microsoft has induced infringement by, with knowledge of the '556 patent, controlling the design and development of, offering for sale, and selling the services of the Azure platform with the knowledge and specific intent that its customers will use the Azure platform to infringe the '556 patent by executing the system operations and utilizing the system components to perform dynamic resource management of the pool of Azure processing resources on behalf of application programs of customers of the Azure platform cloud computing services.

225. Second, on information and belief, Microsoft has, with knowledge of the '556 patent, induced infringement by its customers through the dissemination of promotional, marketing, and tutorial materials relating to the Azure platform with the knowledge and specific

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

intent that its customers will use the Azure platform to infringe the '556 patent by executing the system operations and utilizing the system components to perform dynamic resource management of the pool of Azure processing resources on behalf of application programs of customers of the Azure platform cloud computing services.

226. Third, on information and belief, Microsoft, with knowledge of the '556 patent, has induced infringement by its customers through the creation and online posting of tutorial and "how-to" materials for the Azure platform and/or its individual components in the United States with the knowledge and specific intent that its customers will use the Azure platform to infringe the '556 patent by executing the system operations and utilizing the system components to perform dynamic resource management of the pool of Azure processing resources on behalf of application programs of customers of the Azure platform cloud computing services.

227. Fourth, on information and belief, Microsoft has, with knowledge of the '556 patent, induced infringement through the distribution of other instructional materials, product manuals, and technical materials with the knowledge and the specific intent to encourage and facilitate its customers' infringing use of the Azure platform.

228. Microsoft has engaged in the above activities with knowledge of the '556 patent and with the specific intent to encourage and cause infringement by its customers, as shown by the allegations set forth above.

229. Microsoft has contributed to, or has threatened to contribute to, the infringement by its customers of the '556 patent by, without authority, selling and offering to sell within the United States cloud computing services and customer support services for practicing the claimed invention of the '556 patent, including at least the Azure platform as a whole and/or the individual components of the Azure platform. When, for example, the Azure platform is used by Microsoft's

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

customers for the various cloud computing services Microsoft offers, the Azure system operations and system components are used to perform the claimed dynamic resource management of the pool of Azure processing resources on behalf of application programs of customers of the Azure cloud platform, thereby infringing the '556 patent. In such settings, the Azure platform and/or its individual components, supplied by Microsoft, constitute at least a material part of the claimed invention of the '556 patent.

230. Microsoft's infringement of the '556 patent has injured ThroughPuter in its business and property rights.

231. Microsoft's infringement of the '556 patent has been and is deliberate and willful and constitutes egregious misconduct. Upon information and belief, despite actual knowledge of the '556 patent and numerous related patents and applications since at least May 2015, Microsoft continued to develop and offer its infringing products and services. In developing and offering its products and services, Microsoft has been willfully blind to this ongoing infringement.

232. Pursuant to 35 U.S.C. § 284, ThroughPuter is entitled to recover monetary damages for the injuries arising from Microsoft's willful infringement in an amount to be determined at trial. Microsoft's infringement of the '556 patent has caused irreparable harm to ThroughPuter and will continue to cause such harm unless and until Microsoft's infringing activities are enjoined by this Court.

233. Microsoft's infringement of the '556 patent is exceptional and entitles ThroughPuter to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

PRAYER FOR RELIEF

WHEREFORE, ThroughPuter respectfully requests that the Court enter judgment against Microsoft as follows:

A. An adjudication that Microsoft has infringed one or more claims of the '948 and '556 patents;

B. An order permanently enjoining Microsoft from further infringement of the '948 and '556 patents;

C. An award of damages pursuant to 35 U.S.C. § 284;

D. An order that the damages award be increased up to three times the actual amount assessed, pursuant to 35 U.S.C. § 284;

E. An award to ThroughPuter of its costs, pre- and post-judgment interest, and reasonable expenses to the fullest extent permitted by law;

F. A declaration that this case is exceptional pursuant to 35 U.S.C. § 285, and an award of attorneys' fees and costs; and

G. An award to ThroughPuter of such other and further relief as this Court deems just and proper.

DEMAND FOR JURY TRIAL

Pursuant to Rule 38(b) of the Federal Rules of Civil Procedure, ThroughPuter hereby demands a trial by jury on all issues so triable.

LOWE GRAHAM JONES PLLC
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
206.381.3300 • F: 206.381.3301

DATED this 13th day of April, 2022.

LOWE GRAHAM JONES PLLC

s/Lawrence D. Graham
Lawrence D. Graham, WSBA No. 25402
Graham@LoweGrahamJones.com
1325 Fourth Avenue, Suite 1130
Seattle, Washington 98101
T: 206.381.3300
F: 206.381.3301

Attorneys for Plaintiff ThroughPuter, Inc.